



Power Platform App in a Day

Module 2: Common Data Service for Apps

Hands-on Lab Step-by-Step

September 2018

Contents

Common Data Service for Apps	1
<i>Lab Prerequisites</i>	<i>1</i>
<i>Before you begin.....</i>	<i>1</i>
<i>Overview.....</i>	<i>2</i>
<i>Goals for this lab</i>	<i>2</i>
Exercise 1: Exploring the Common Data Service	3
<i>Task 1: Explore standard entities</i>	<i>3</i>
<i>Task 2: Explore standard option sets</i>	<i>7</i>
Exercise 2: Custom Entities and Fields.....	8
<i>Task 1: Create a custom entity</i>	<i>8</i>
<i>Task 2: Create custom fields.....</i>	<i>10</i>
<i>Task 3: Create a calculated field.....</i>	<i>16</i>
<i>Task 4: Create a business rule</i>	<i>20</i>
Exercise 3: Connect the data from the Canvas App	25
<i>Task 1: Add CDS entity as a data source to the app</i>	<i>25</i>
<i>Task 2: Create the edit form.....</i>	<i>29</i>
<i>Task 3: Configure the title field</i>	<i>35</i>
<i>Task 4: Configure the price field</i>	<i>39</i>
<i>Task 5: Configure the approval field</i>	<i>41</i>
<i>Task 6: Configure the Comment field</i>	<i>44</i>
<i>Task 7: Configure the Requested By field</i>	<i>45</i>
<i>Task 8: Configure the requested date field.....</i>	<i>46</i>
<i>Task 9: Add a button to submit the form.....</i>	<i>47</i>
<i>Task 10: Test the form</i>	<i>50</i>
<i>Task 11: Verify a new item was added to the Device Order entity</i>	<i>53</i>
<i>Task 12: [Optional] Navigate to confirmation screen after the Form submit is successful</i>	<i>53</i>
Lab survey.....	58
References	58

Common Data Service for Apps

Lab Prerequisites

This is the second lab in a five-part series covering PowerApps Canvas Apps, Common Data Service for Apps, PowerApps Model-driven Apps and Microsoft Flow. The assumption is that you have successfully completed the initial part of setting up an environment as described in the overview document – “00-AppInADay Lab Overview.pdf”.

If you have not completed building the PowerApps Canvas App in module 1, you can use the partially completed version of the lab package in the “\Completed\Module1” folder. Follow the instructions in the document “Importing Module 1 Completed” before proceeding with this module.

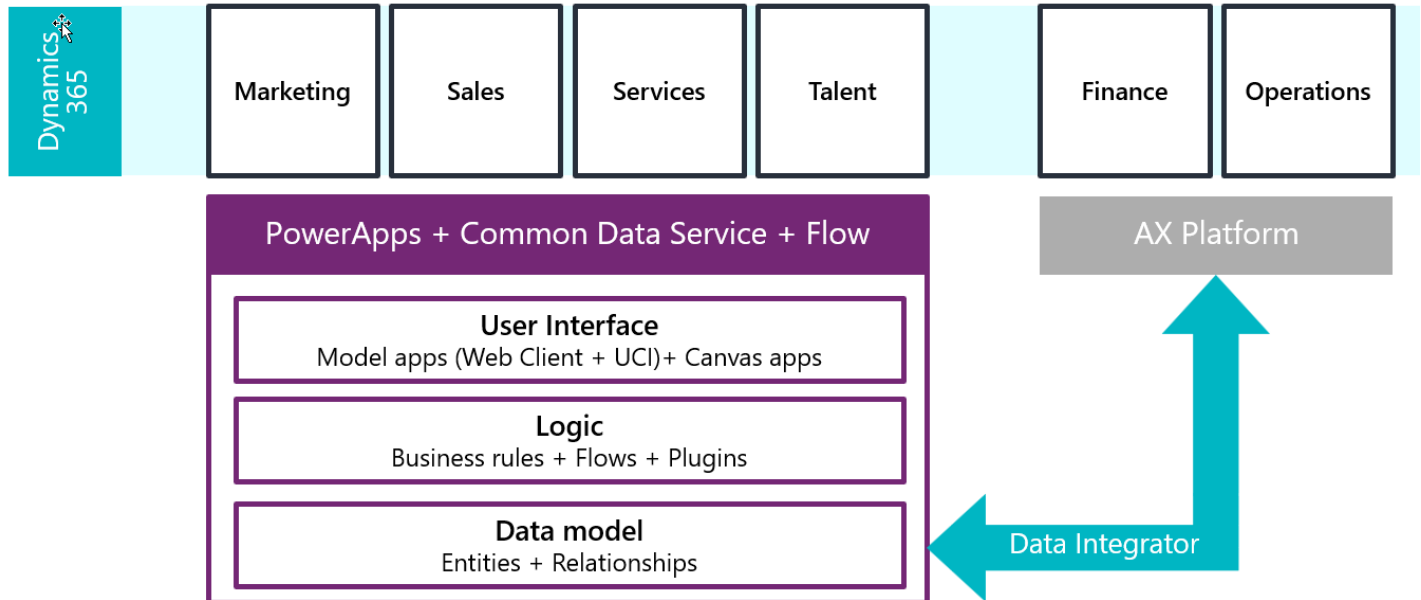
Before you begin

You must be connected to the internet.

1. **System requirements:** At a minimum, a computer with 2-cores and 4GB RAM running Windows 7+ or MacOS.
2. **Have a Test Environment with permission to create Common Data Service database:** You should have gone through the steps to create a new environment using the Admin center. In this lab, you will create a database in this environment if you haven't already created one.
3. **Sign-in to PowerApps:** Go to <http://powerapps.com> and **sign in** with the same account you used to complete the first lab. Make sure you switch to the environment where you created the app.

Overview

The [Common Data Service for Apps \(CDS for Apps\)](https://docs.microsoft.com/powerapps/common-data-service/overview) adds data storage and modeling capabilities to PowerApps that is scalable and easy to provision. In addition to allowing for the creation of custom entities, CDS for Apps contains a Common Data Model (CDM) currently consisting of 100+ standard entity definitions. You can find the current CDM schema at <https://github.com/Microsoft/CDM> and you can browse the CDM using the CDM Visual Entity Navigator located here <https://microsoft.github.io/CDM/>. You can read more about the CDM here <https://docs.microsoft.com/powerapps/common-data-model/overview>



Goals for this lab

After this lesson you will be able to:

- Provision a Common Data Service database.
- Create a custom entity and add custom fields to it.
- Use the PowerApps Form control to populate the entity table
- View the entity data in the entity table.
- Create a calculated field.
- Implement a server-side business rule.



The time to complete this lab is **[45]** minutes.

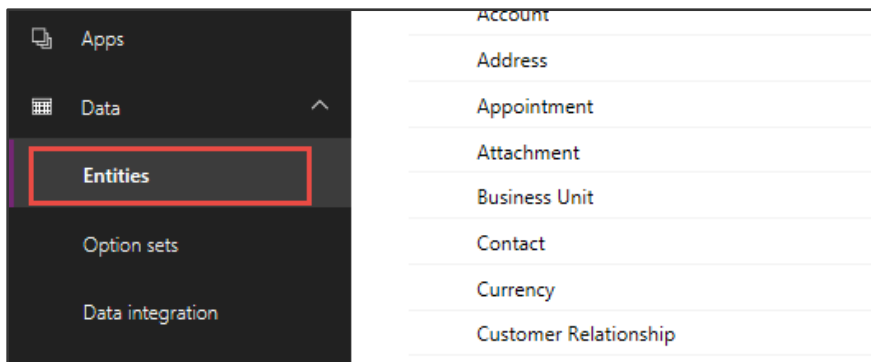
Exercise 1: Exploring the Common Data Service

In this exercise, you will explore the Common Data Service standard entities.

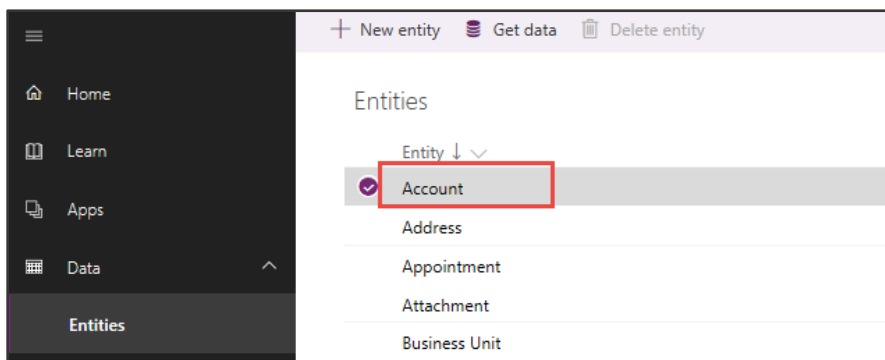
Task 1: Explore standard entities

In this task, you will explore Common Data Service standard entities.

1. Navigate to <http://web.powerapps.com> and in the left pane, expand **Data** and select the **Entities** option.



2. This will bring up the list of entities in this database instance. Click on a few of the standard entities (for example, **Account**) to get familiar with some of the features of an entity.



For detailed documentation on CDS entities, see <https://docs.microsoft.com/en-us/powerapps/developer/common-data-service/reference/about-entity-reference>

Fields:

An entity has a list of fields. Each field has a data type.

Fields of standard entities cannot be edited or removed, but you can extend existing entities by adding additional custom fields.

For a list of supported data types, see <https://docs.microsoft.com/dynamics365/customer-engagement/customize/types-of-fields>. Please note that a few of the field names have changed slightly in how they present in the list of choices. For example, Single Line of Text is just Text.

Entities > Account

Fields Relationships Business rules Views Keys Data

Display name ↓		Data type ↓	Type ↓	Required ↓	Searchable ↓
Account Name	...	Text	Standard	✓	✓
Account Number	...	Text	Standard		✓
Account Rating	...	Option Set	Standard		
Address 1	...	Multiline Text	Standard		✓
Address 1: Address Type	...	Option Set	Standard		✓
Address 1: City	...	Text	Standard		✓
Address 1: Country/Region	...	Text	Standard		✓

Relationships:

Allows you to manage relationships between entities. Relationships supported are One to Many (1:N), Many to One (N:1) and Many to Many (N:N). Relationships also define the behavior that happens when actions occur on the primary record in a 1:N relationship. For example, if the parent record is deleted you can configure the relationship behavior so that all child records are also deleted or simply remove the reference.

Entities > Account

Fields **Relationships** Business rules Views Keys Data

Display name ↓		Related entity ↓	Relationship type ↓	Type ↓
Company Name	...	Contact	One-to-many	System
Connected From	...	Connection	One-to-many	System
Connected To	...	Connection	One-to-many	System
Created By	...	User	Many-to-one	System

Business Rules:

Building a Business Rule is like building a flowchart where you can define conditions and actions.

More about Business Rules <https://docs.microsoft.com/dynamics365/customer-engagement/customize/create-business-rules-recommendations-apply-logic-form>

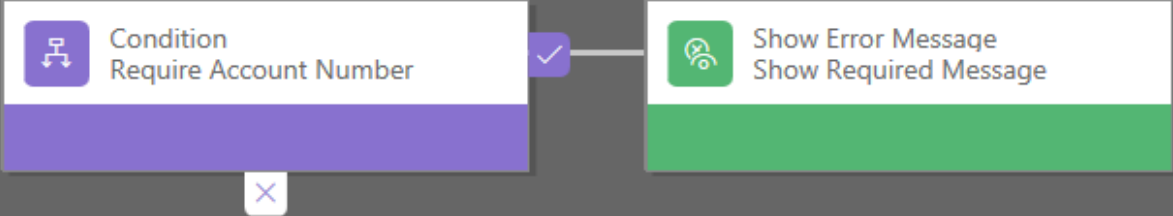
Entities > Account

Fields Relationships **Business rules** Views Keys Data

Name ↓ ▾	Status ▾	Scope ▾
Require Account Number <input checked="" type="checkbox"/>	Activated	Entity

Account: Require Account Number ▾

+ Add ✂ Cut 📄 Copy 📄 Paste 🗑 Delete 📷 Snapshot



Views:

Views will let you define how list records are shown in the app. You can create multiple custom views, each having their own filtering and sorting criteria. For example, you could create a view to see only the records created in the last week and another one to see records that haven't been updated in a year. Create views to make the application users more productive in filtering their data.

Entities > Account

Fields Relationships Business rules **Views** Keys Data

Name ↓ ▾	Type ▾
Account Advanced Find View <input checked="" type="checkbox"/>	... Standard
Account Associated View <input checked="" type="checkbox"/>	... Standard
Account BulkOperation View <input checked="" type="checkbox"/>	... Standard
Account List Member View <input checked="" type="checkbox"/>	... Standard

Keys:

Allows you to view the lookup keys for the entity. Keys can contain multiple fields to define a composite key. Keys enforce uniqueness, so they should not be used when there is a need to store duplicate values of fields used.

Entities > Account

Fields Relationships Business rules Views **Keys** Data

Display name ↓ ▾ Name ▾ Number

We didn't find anything to show here

Account Number Key

Display name *

Account Number Key

Name *

crf61_ AccountNumberKey

Fields *

☐ Account Name

☒ Account Number

☐ Email

Data:

You can view and search the data in the entity table. This gives you a quick way to see some of the data for the entity without having to jump into a specific Canvas or Model-driven app.

Refresh data Get data ▾ Export data Open in Excel Delete entity Settings My Active Accounts

Entities > Account

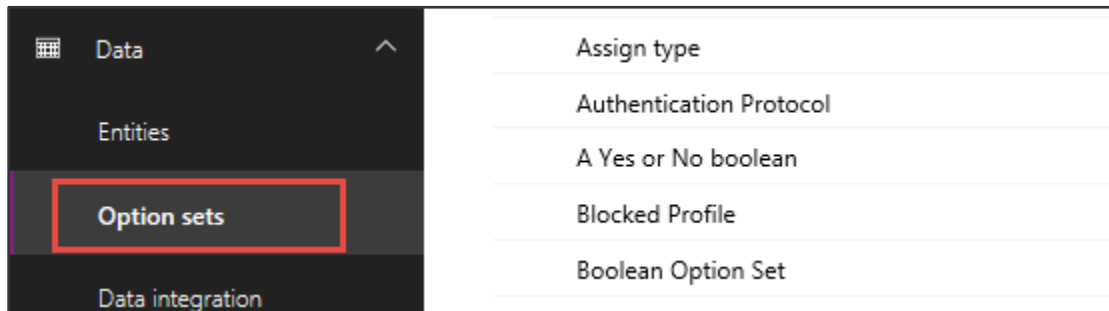
Fields Relationships Business rules Views Keys **Data**

ACCOUNT NAME	MAIN PHONE	ADDRESS 1: CITY	PRIMARY CONTACT
A. Datum Corporation (sample)	555-0158	Redmond	Rene Valdes (sample)
Adventure Works (sample)	555-0152	Santa Cruz	Nancy Anderson (sam..
Alpine Ski House (sample)	555-0157	Missoula	Paul Cannon (sample)
Blue Yonder Airlines (sample)	555-0154	Los Angeles	Sidney Higa (sample)
City Power & Light (sample)	555-0155	Redmond	Scott Konersmann (sa..

Task 2: Explore standard option sets

Just like standard entities, the Common Data Service includes a set of standard **Option Sets**. You can also create custom **Option Sets**. Later in this lab, we will create a custom **Option Set** called **ApprovalStatus** to set the approval status of a device order.

1. Select **Option Sets**.



2. Scroll through the standard **Option Sets**.

+ New option set		Search
Option sets		
Option set ↓ ▾	Type ▾	
Activity Type	System	...
Approve Article	System	...
Assign type	System	...
Authentication Protocol	System	...
A Yes or No boolean	System	...
Blocked Profile	Standard	...
Boolean Option Set	System	...
Category	Standard	...
Channel Activities	System	...
Component State	System	...
Component Type	System	...
Convert Activity	System	...

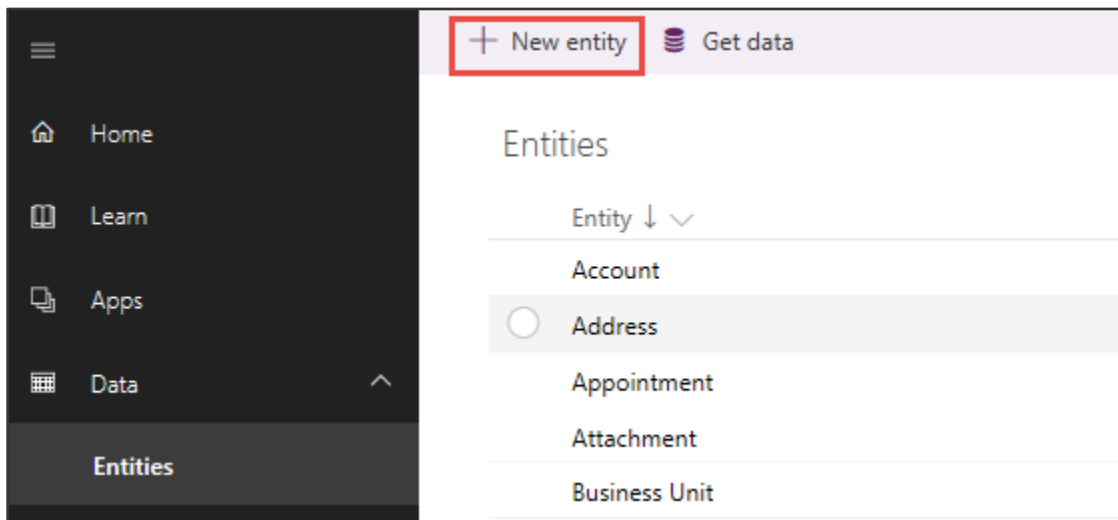
Exercise 2: Custom Entities and Fields

In this exercise, you will create a new custom entity named Device Order, add fields necessary to track the device requests. You will also create a server-side Business Rule that will default the estimated ship date.

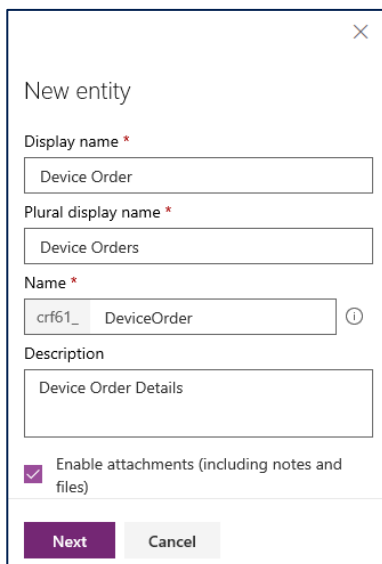
Task 1: Create a custom entity

In this task, you will create a custom entity to store device order requests.

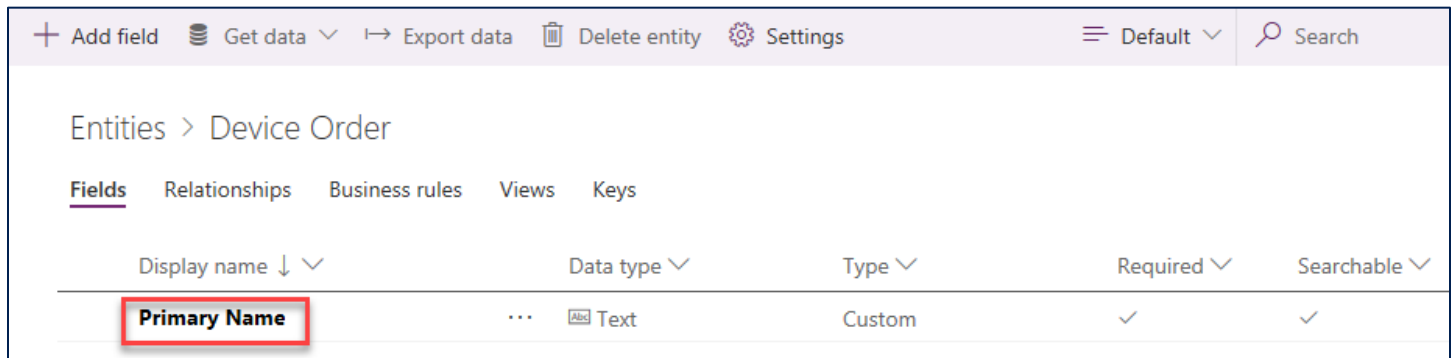
1. Select **Entities** in the left pane and click **New Entity** in the upper left corner of the page.



2. Enter **Device Order** for Display Name, **DeviceOrder** for Name, **Device order details** for Description. Check the Enable attachments since this will allow creating notes on the device order. Then click **Next**. The Plural display name will be populated automatically but if needed you can override the name provided. The plural name is used by the system by default anytime a set of the records are shown.

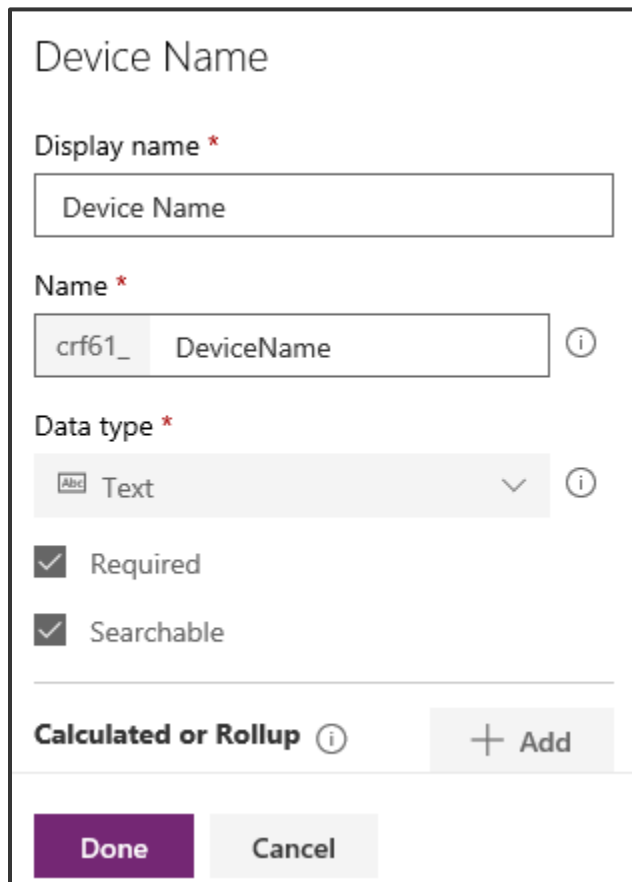
A screenshot of the 'New entity' dialog box in Microsoft Dynamics 365. The dialog has a title bar with a close button. It contains several input fields: 'Display name' with the value 'Device Order', 'Plural display name' with the value 'Device Orders', 'Name' with the value 'crf61_ DeviceOrder' (with a help icon), and 'Description' with the value 'Device Order Details'. Below these fields is a checkbox labeled 'Enable attachments (including notes and files)' which is checked. At the bottom are two buttons: 'Next' (highlighted in purple) and 'Cancel'.

- After you click Next you will be able to change the primary attribute (labeled as Primary Name) and also create other custom Fields prior to actually saving the entity and causing the system to create it. The primary attribute defaults to being named Primary Name, for some scenarios that might not be the best label and you can customize it if needed. The primary attribute however is always a Text field, that is not changeable.
- Click on Primary Name.



Entities > Device Order				
Fields	Relationships	Business rules	Views	Keys
Display name ↓	Data type ↓	Type ↓	Required ↓	Searchable ↓
Primary Name	Text	Custom	✓	✓

- Change Display name to Device Name, Name to DeviceName, and click Done.



Device Name

Display name *

Device Name

Name *

crf61_ DeviceName

Data type *

Text

☒ Required

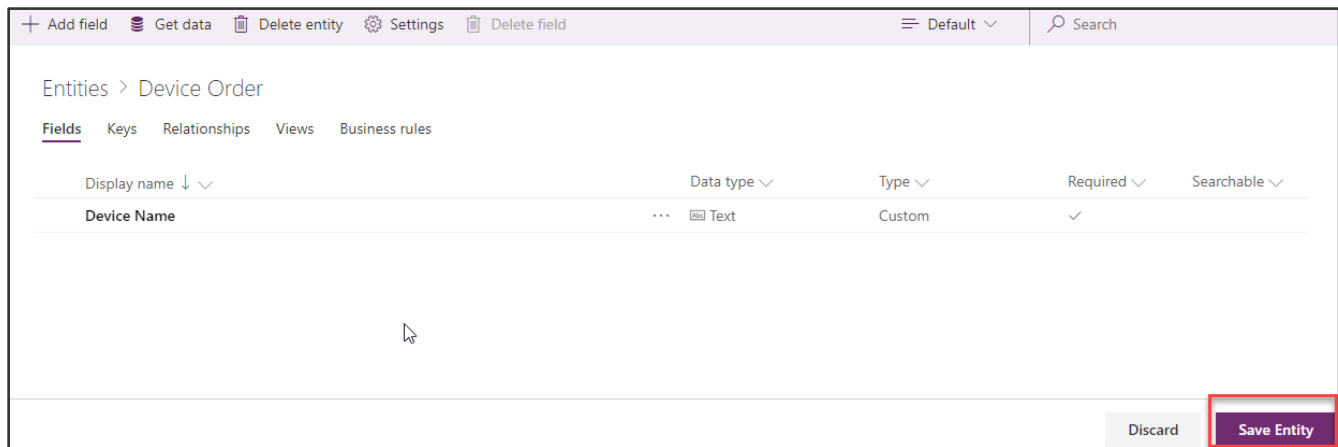
☒ Searchable

Calculated or Rollup ⓘ

+ Add

Done Cancel

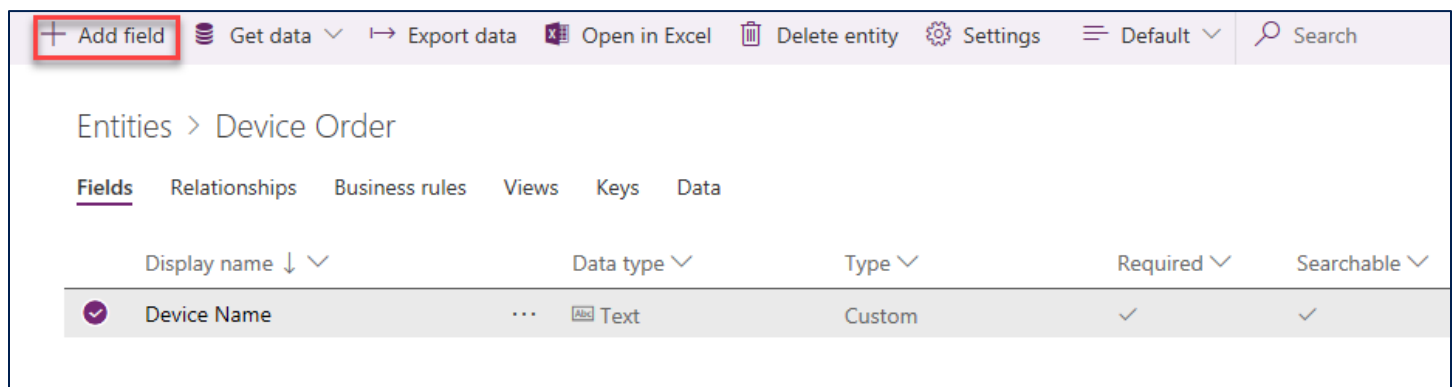
- Click Save Entity.



Task 2: Create custom fields

In this task, you will create custom fields for the Device Order entity.

1. Click on the **Add field** button to add fields to your custom entity.



2. Enter **Price** for Display Name, select **Currency** for Data Type, make the field **Required** and **Searchable** and click **Advanced Options**.

Note: Currency is a special data type. For each currency field you add, another currency field is added with the prefix "_Base" on the name. This field stores the calculation of the value of the currency field you added and the base currency. For additional information on using the Currency field, see [here](#).

Price

Display name *

Name *

crf61_ Price (i)

Data type *

Currency (i)

☐ Required

☒ Searchable

Calculated or Rollup (i) + Add ▼

Advanced options ▼

Done Cancel

3. Enter **Device Price** for Description, **0** for Minimum Value, **5000** for Maximum Value, and click **Done**.

Price

Description

Device Price

Minimum value

0

Maximum value

5,000

Precision source

Currency precision

IME mode

Auto

Done Cancel

4. Click Add Field again.
5. Enter **Requested By** for Display Name, **RequestedBy** for Name, **Email** for Data type, make the field **Searchable** and click **Done**.

Requested By

Display name *

Requested By

Name *

crf61_ RequestedBy

Data type *

Email

☐ Required

☒ Searchable

6. Repeat the steps 5 and 6 to create the fields listed below.

Display Name	Name	Type
Request Date	RequestDate	Date Only
Approver	Approver	Email
Comments	Comments	Multiline Text
Estimated Ship Date	EstimatedShipDate	Date Only
Approved Date	ApprovedDate	Date Only

7. Now we are going to create the Approval Option Set. Click Add New Field.
8. Enter **Approval Status** for Display Name, **ApprovalStatus** for Name, select **Option Set** for Data type, and select **New Option Set** for Option Set.

Approval Status

Display name *

Approval Status

Name *

crf61_ ApprovalStatus

Data type *

Option Set

Option set *

+ New option set

A Yes or No boolean

9. Change the New Item label to **Approve**
10. Click **Add new item**.

Approval Status

Display name *

Approval Status

Name *

crf61_ approvalstatus

View more

Items (1)

Approve

Add new item

11. Enter **Reject** and click **Save**.

Items (2)

Approve	...
Reject	...

[Add new item](#)

Save Cancel

12. Click **Done**.

Approval Status

Display name *

Approval Status

Name *

crf61_ ApprovalStatus ⓘ

Data type *

Option Set ⓘ

Option set *

Approval Status ▼

Edit option set

Default value

▼

Done Cancel

13. Click Save Entity.

Entities > Device Order

Fields Relationships Business rules Views Keys Data

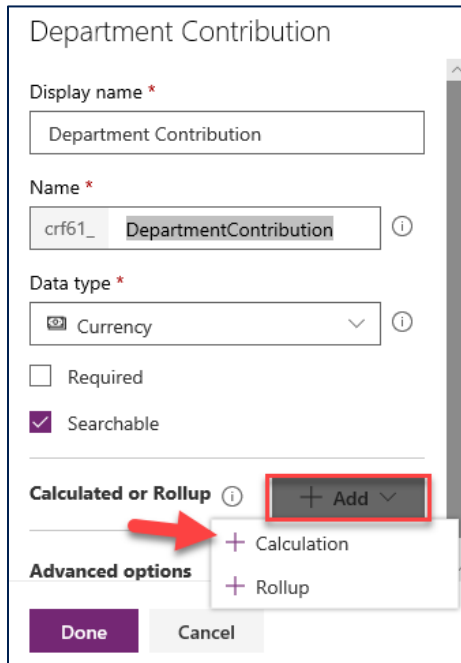
Display name ↓		Data type ↓	Type ↓	Required ↓	Searchable ↓
✓ Approval Status	...	Option Set	Custom		✓
Approved Date	...	Date Only	Custom		✓
Approver	...	Email	Custom		✓
Comments	...	Multiline Text	Custom		✓
Device Name	...	Text	Custom	✓	✓
Estimated Ship Date	...	Date Only	Custom		✓
Price	...	Currency	Custom		✓
Request Date	...	Date Only	Custom		✓
Requested By	...	Email	Custom		✓

Discard Save Entity

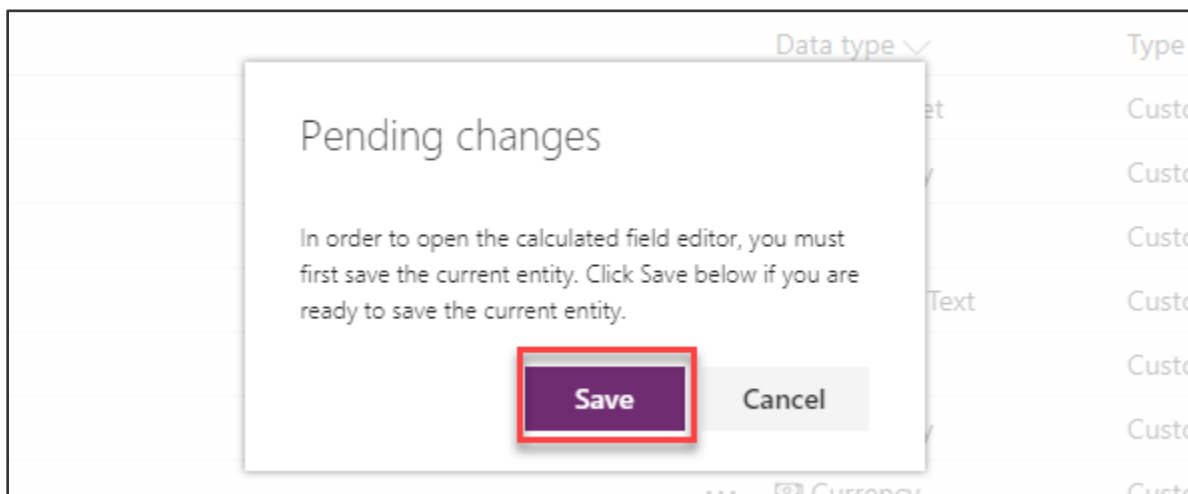
Task 3: Create a calculated field

In this task, you will add a Department Contribution field and set its value to 10% of the price. In our scenario, this is the amount that will come from the department managers budget. Calculated fields are special fields that automatically perform the calculation when the data is retrieved. When you create or modify a calculated field you can set the formula used in the calculation.

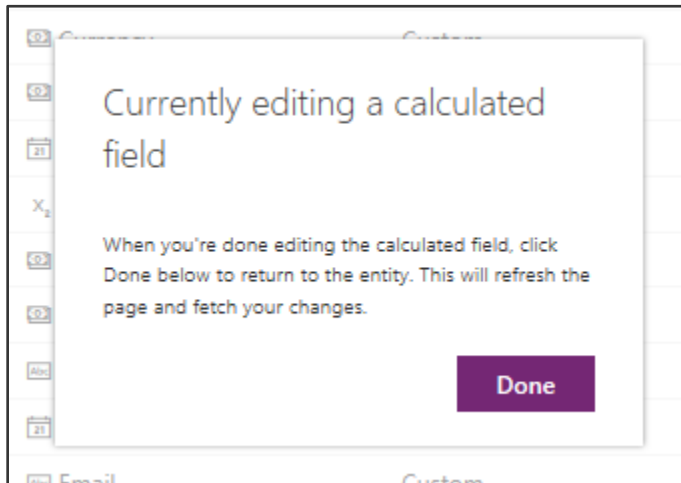
1. In the upper left corner of the screen, click on **Add field** to add fields to your custom entity.
2. Enter **Department Contribution** for Display Name, enter **DepartmentContribution** for Name, select **Currency** for Data Type, click **Add Calculated or Rollup**, and select **Calculation**.



3. Click Save.



4. Click **Done**.



5. Select the **Department Contribution** field from the list of fields, then click **Open calculation**.

Department Contribution

Display name *

Department Contribution

Name *

crf61_ DepartmentContribution ⓘ

Data type *

Currency ⓘ

☒ Searchable

Calculated or Rollup ⓘ

Open calculation ☒

Advanced options ∨

6. Click **Add Action**.

SAVE SAVE AND CLOSE

CALCULATED FIELD

Set Department Contribution

IF...THEN

CONDITION (OPTIONAL)

+ Add condition

ACTION

+ Add action

7. Type price and select the Price field you created.

CALCULATED FIELD

Set Department Contribution

IF...THEN

CONDITION (OPTIONAL)

+ Add condition

ACTION

Set Department Contribution (currency)

= price

crf61_price (currency)

8. Add ***.1** and click the **Check Mark** button.

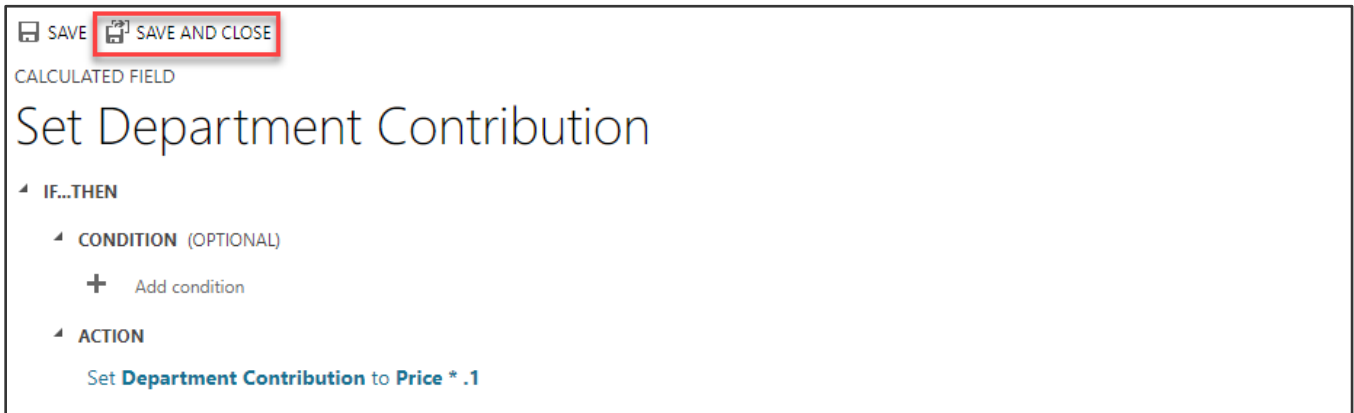
Set Department Contribution (currency)

price * 0.1

= crf61_price * 0.1

The formula contains an invalid reference: price.

✓ ✕

9. Click **Save and Close**.

SAVE SAVE AND CLOSE

CALCULATED FIELD

Set Department Contribution

IF...THEN

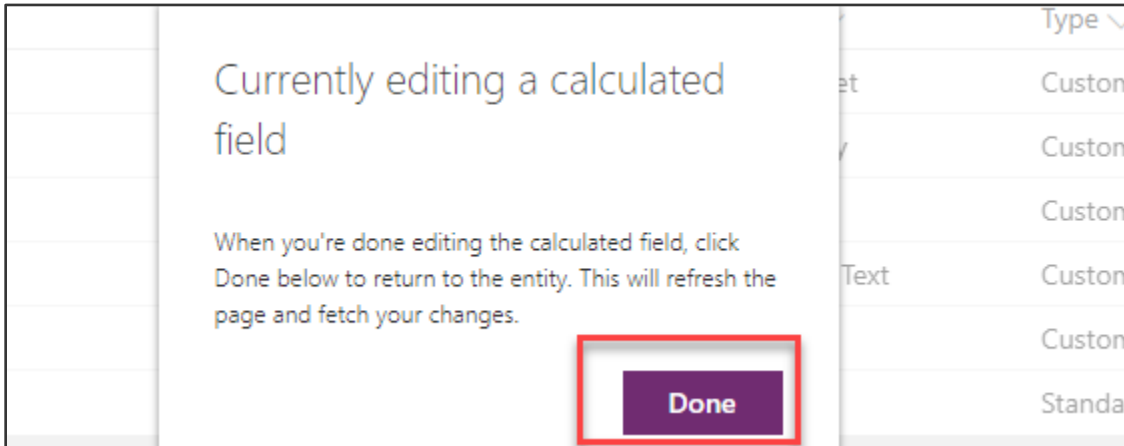
CONDITION (OPTIONAL)

+ Add condition

ACTION

Set **Department Contribution** to **Price * .1**

10. Click Done.



Currently editing a calculated field

When you're done editing the calculated field, click Done below to return to the entity. This will refresh the page and fetch your changes.

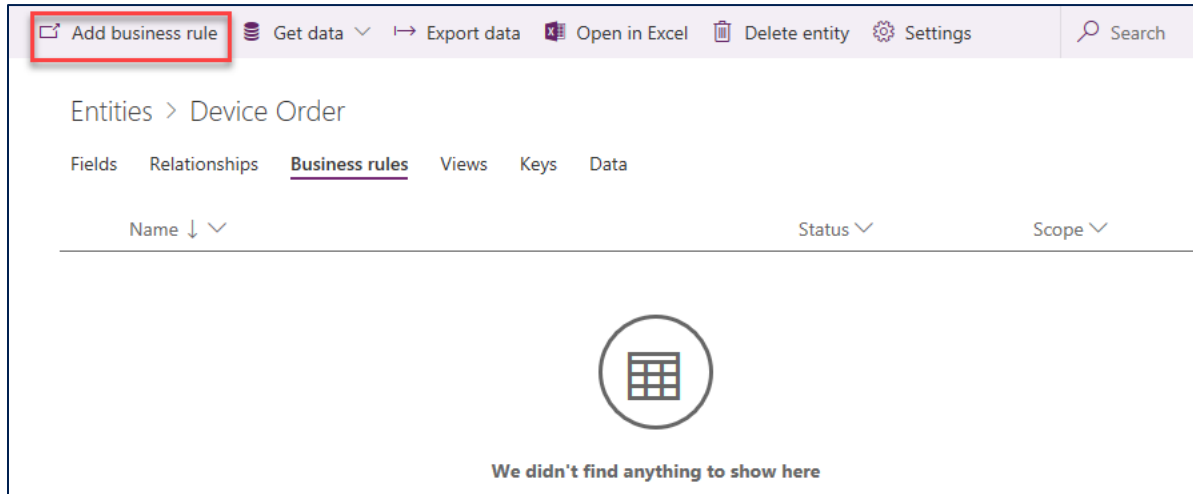
Done

Type
Custom
Custom
Custom
Text
Custom
Custom
Standard

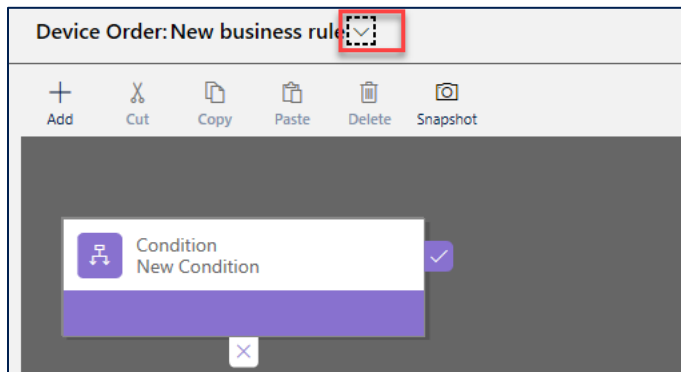
Task 4: Create a business rule

In this task, you will create a **Business Rule** that will set the Estimated Delivery Date to 14 days after approval of the order.

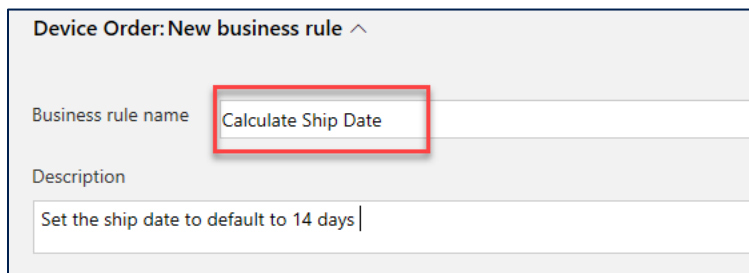
1. Select the **Business Rules** tab and click **Add Business Rule**.



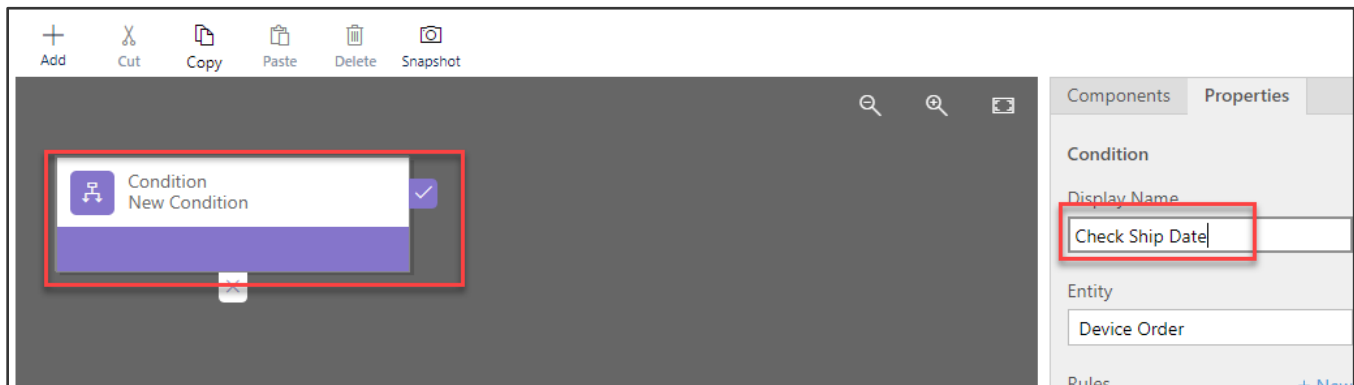
2. Click the arrow to **Show Details**.



3. Change the Name to **Calculate Ship Date** and click the arrow to Hide Details.

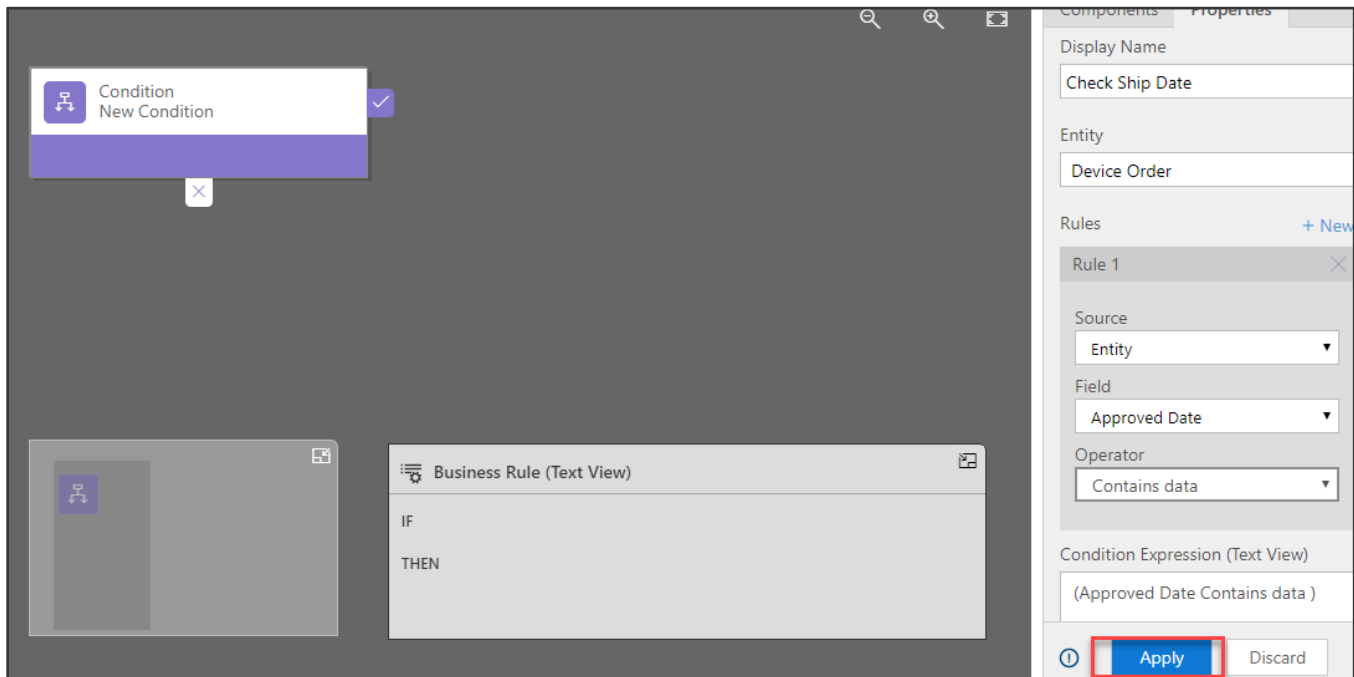


4. Select the **Condition**, change the name to **Check Ship Date**.

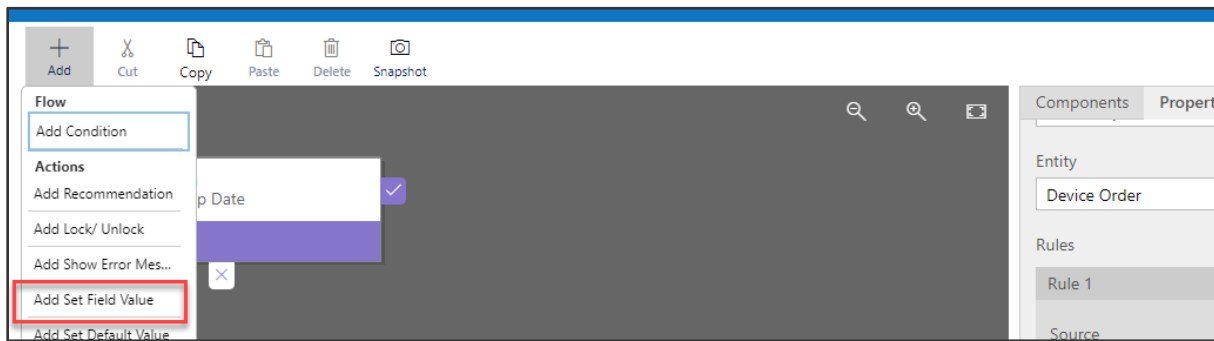


5. In the **Rule 1** section select **Entity** for Source, **Approved Date** for Field, **Contains Data** for Operator and click **Apply**.

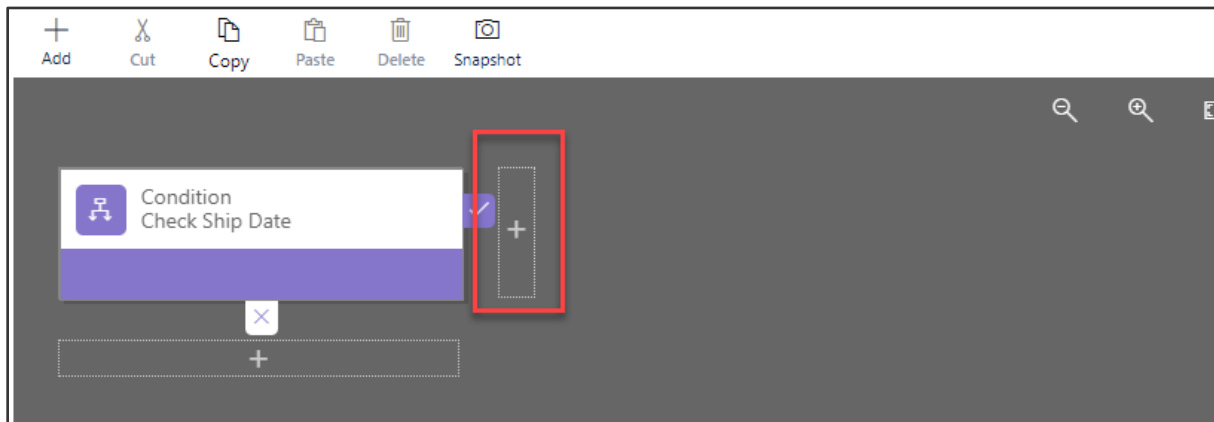
Note: You may need to scroll down to the bottom of all scroll bars to see the Apply button. You must click Apply after any change to the properties otherwise they will revert to the prior value.



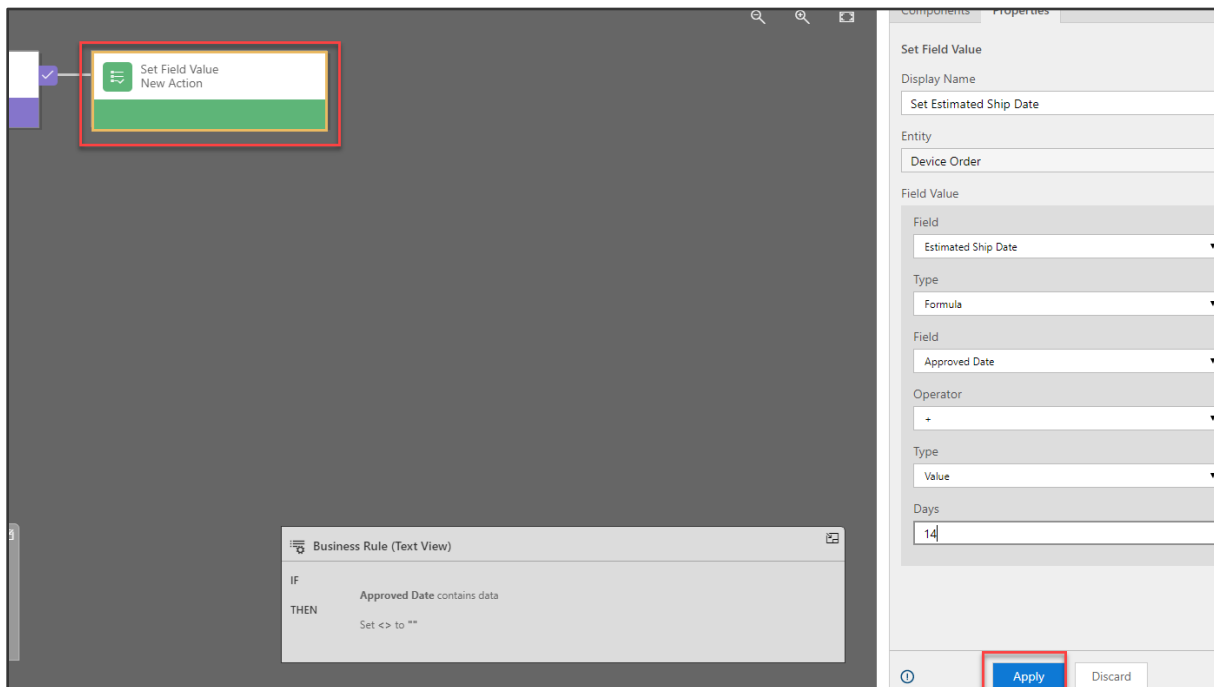
6. Click **Add**, select **Set Field Value**.

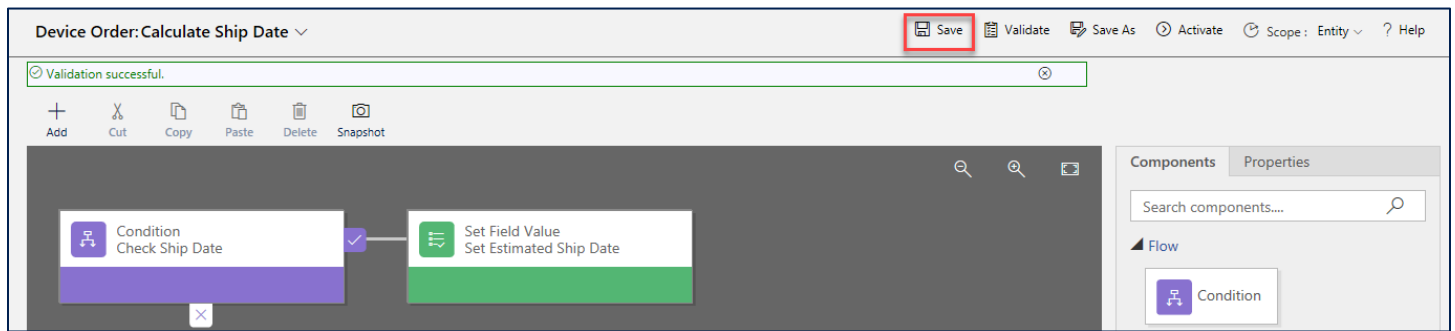
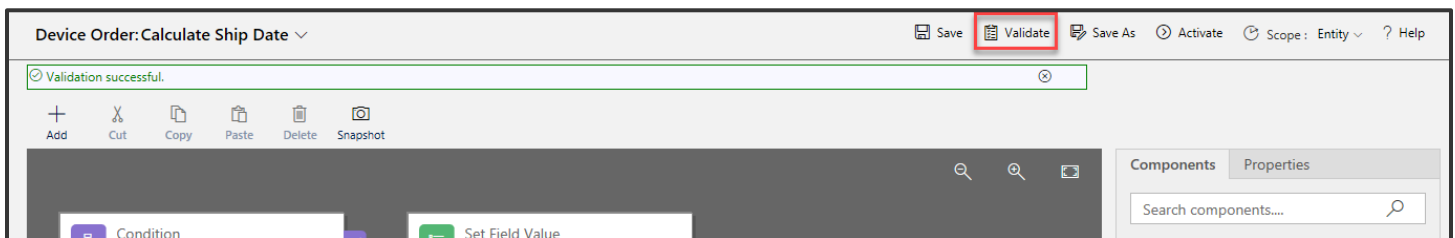
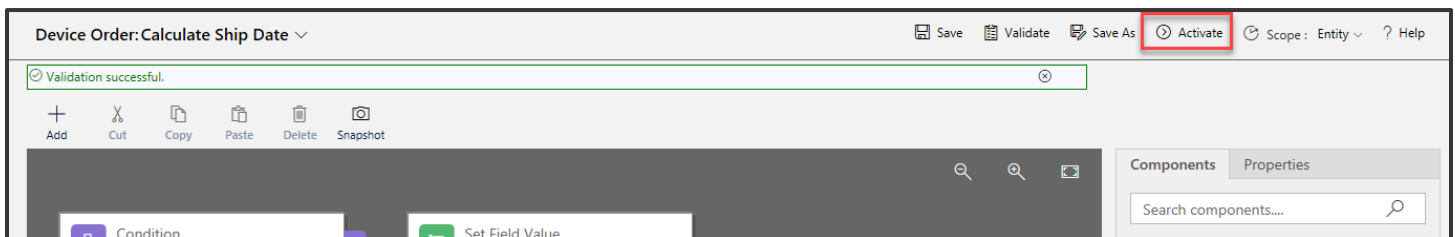


7. Select the True side of the condition.

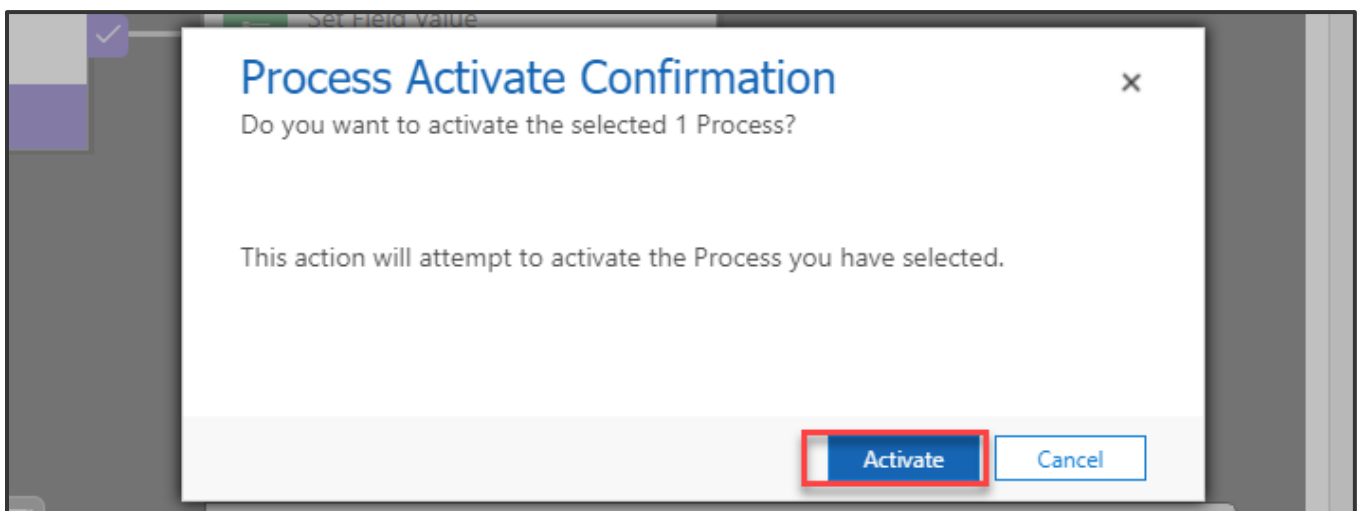


8. Enter **Set Estimated Ship Date** for Display Name, select **Estimated Ship Date** for Field, **Formula** for Type, **Approved Date** for Field, **+** for Operator, **Value** for Type, **14** for Days, and click **Apply**.

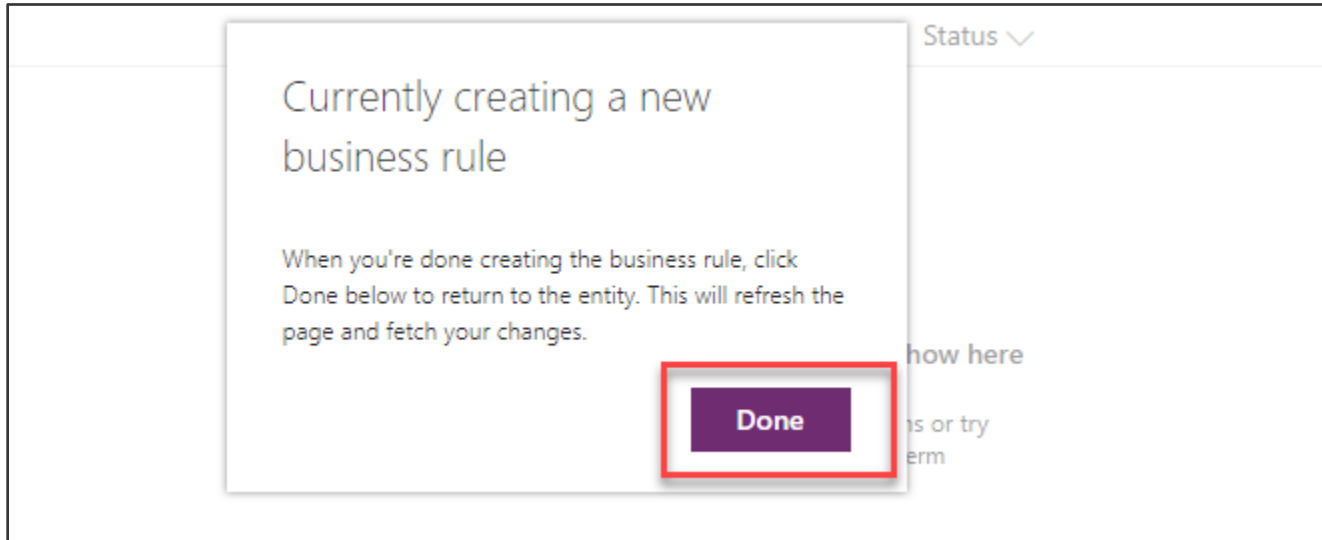


9. Click **Save**.10. Click **Validate**.11. Click **Activate**.

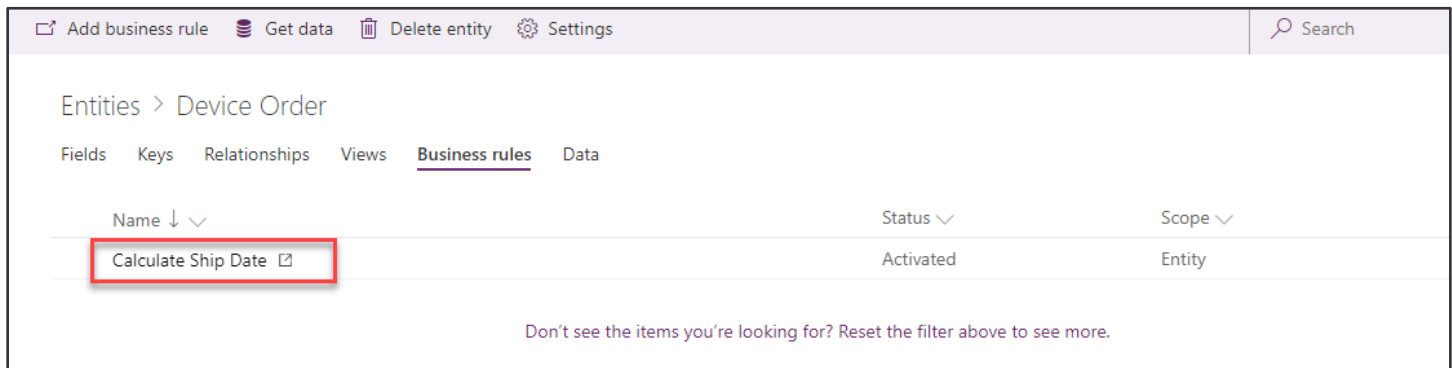
12. Confirm activation. Business rules only execute when they are activated. In the future to make changes to rules you deactivate them, make the change, and then re-activate the rule.



13. Close the process editor browser window or tab.
14. Click **Done**. The list should refresh showing the Business Rule you just created.



15. Your Common Data Service will have one Business Rule.



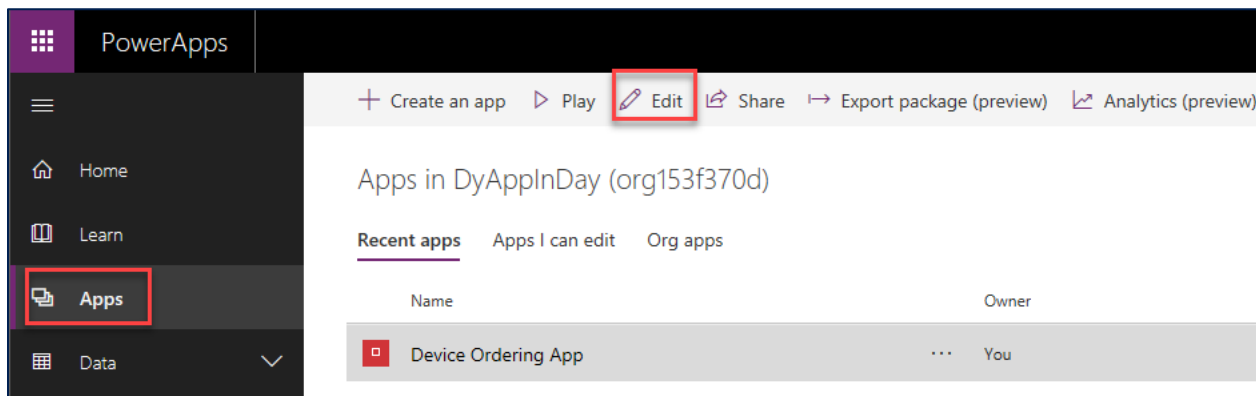
Exercise 3: Connect the data from the Canvas App

Now that you have created the entity to store device order requests let's connect your Device Ordering Canvas app to this entity and add a form to submit device approval requests.

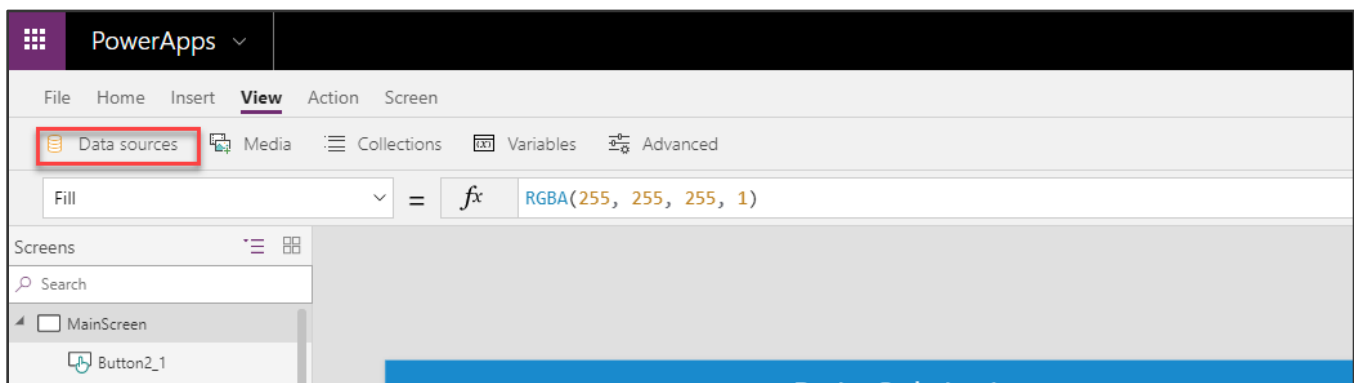
Task 1: Add CDS entity as a data source to the app

Open the device ordering app. Make sure you are opening the version of the app that is in the newly created environment that has the CDS database instance.

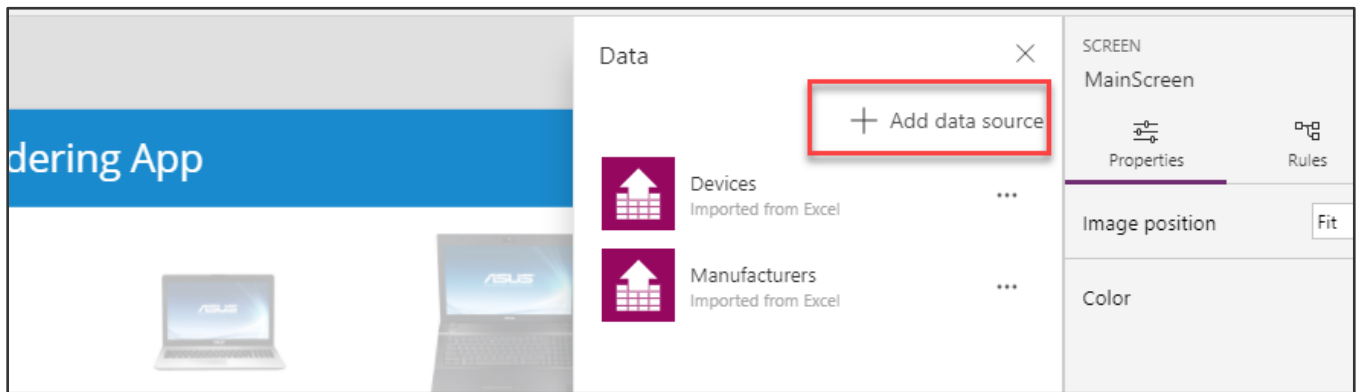
1. Navigate to the Device Order App you created in Module 1, select the App and click **Edit**.



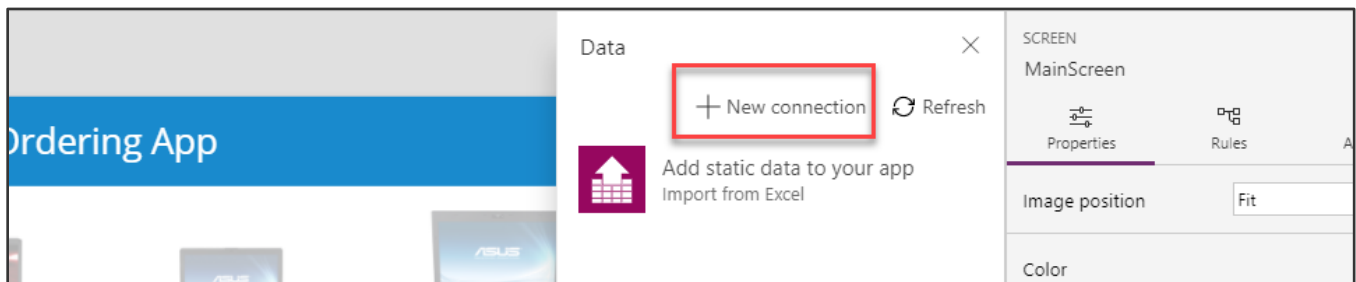
2. Select the **View** tab and then click **Data sources** to display the Data sources property pane to the right of the canvas.



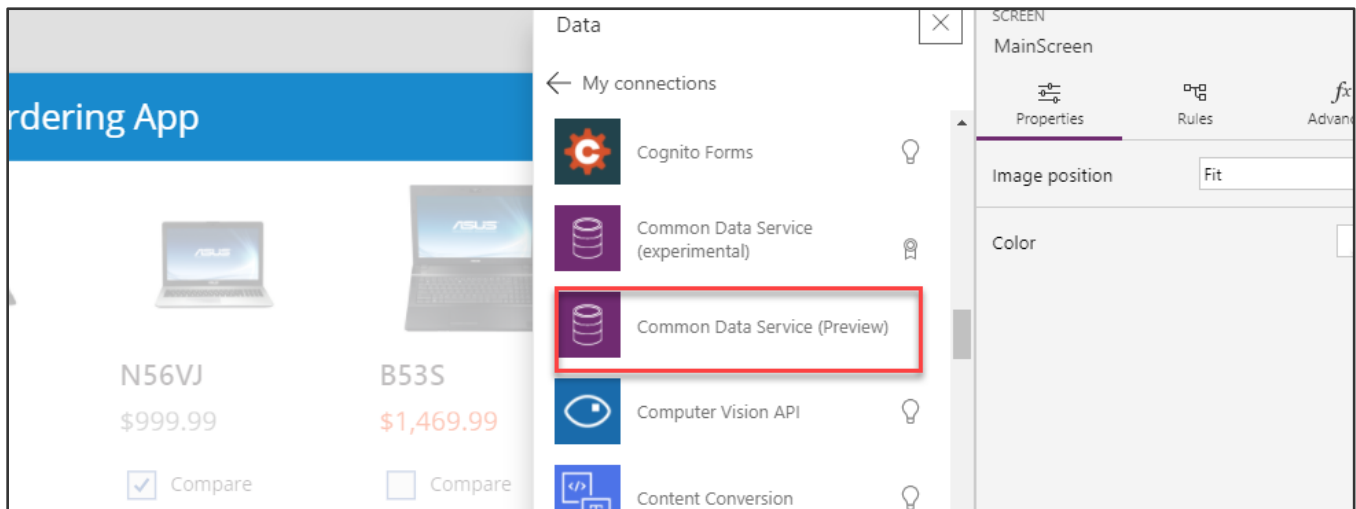
3. Click **Add Data Source**.



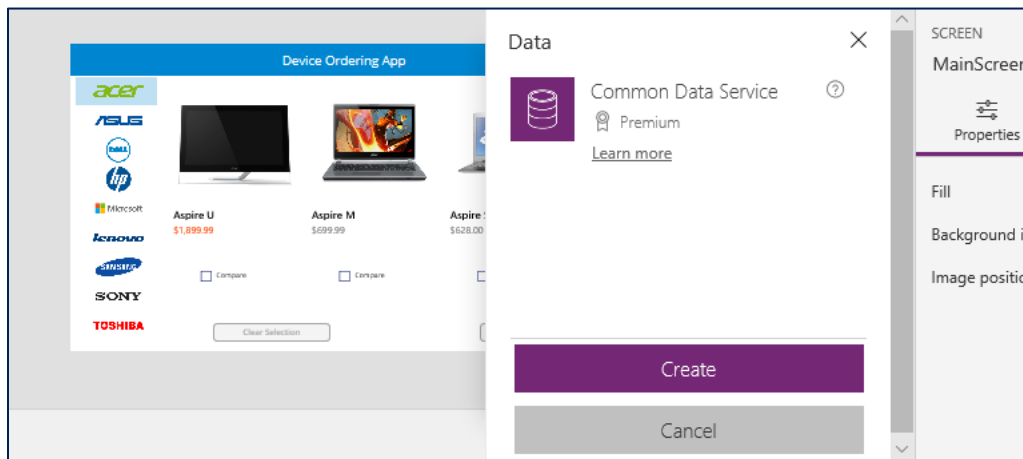
4. Click **New Connection**.



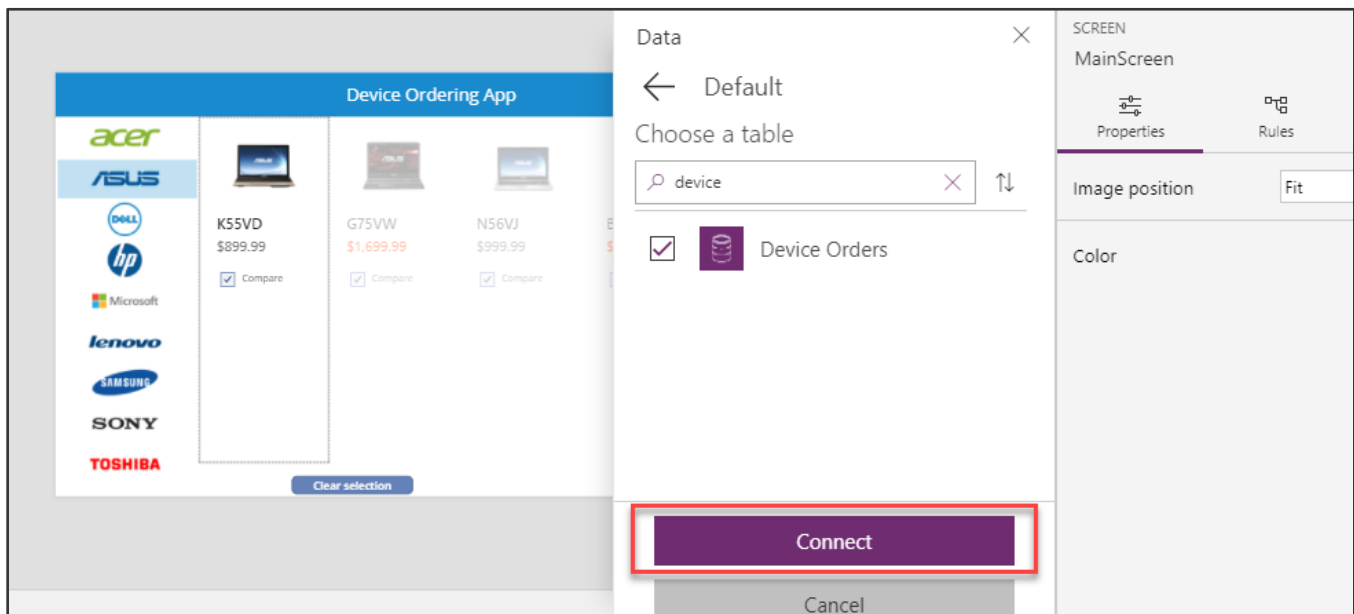
5. Select **Common Data Service**. Double check that you *did not* choose the "experimental" connector.



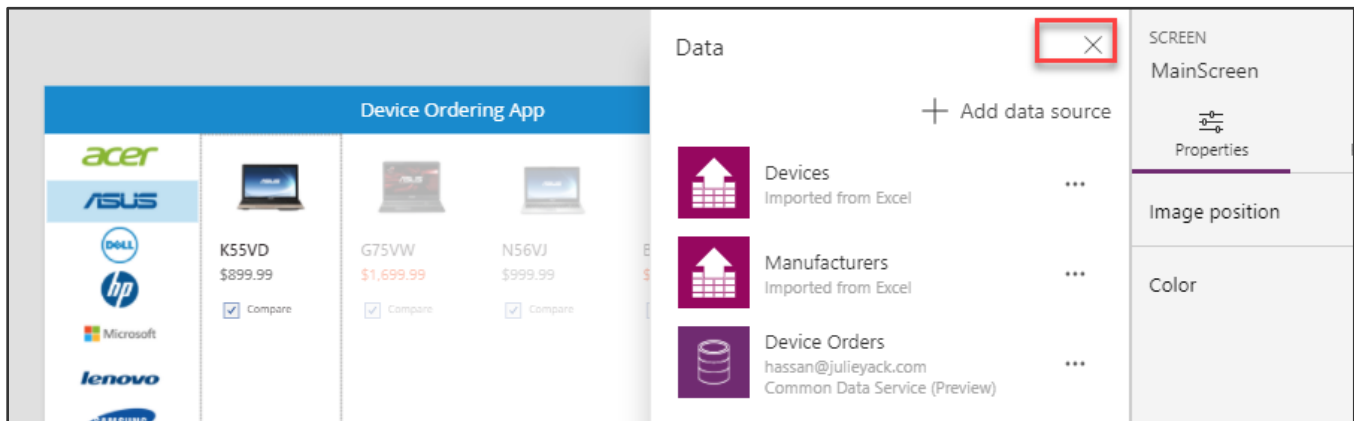
6. Click **Create**.



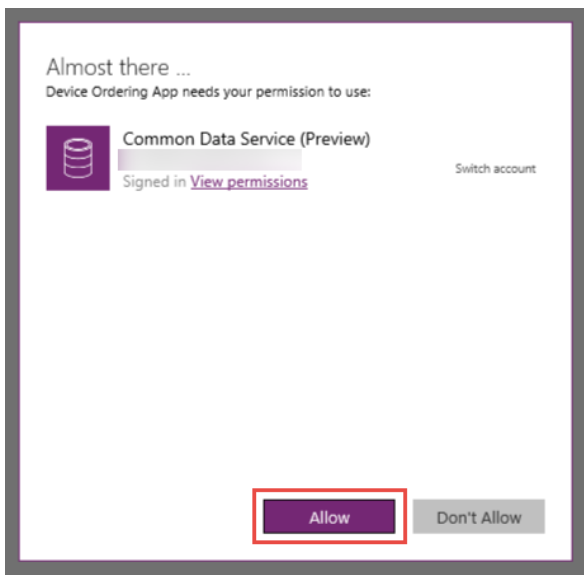
7. Type device in the search text box, this will filter the list of all entities down to just those that contain device. Select **Device Orders** and click **Connect**.



8. Close the Data pane.

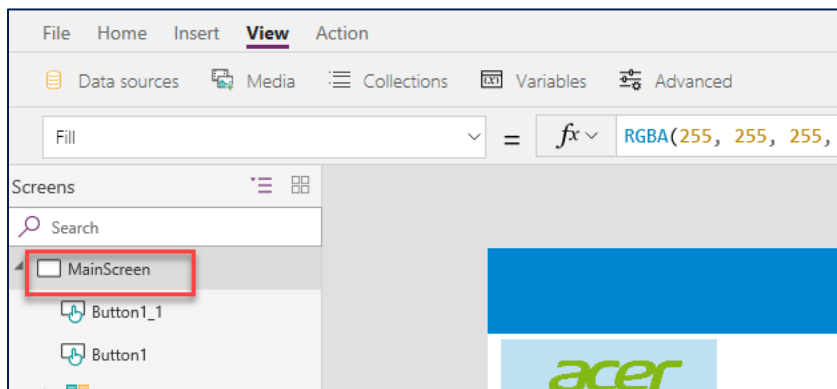


9. If prompted, click Allow to authorize the App to access the Common Data Service.

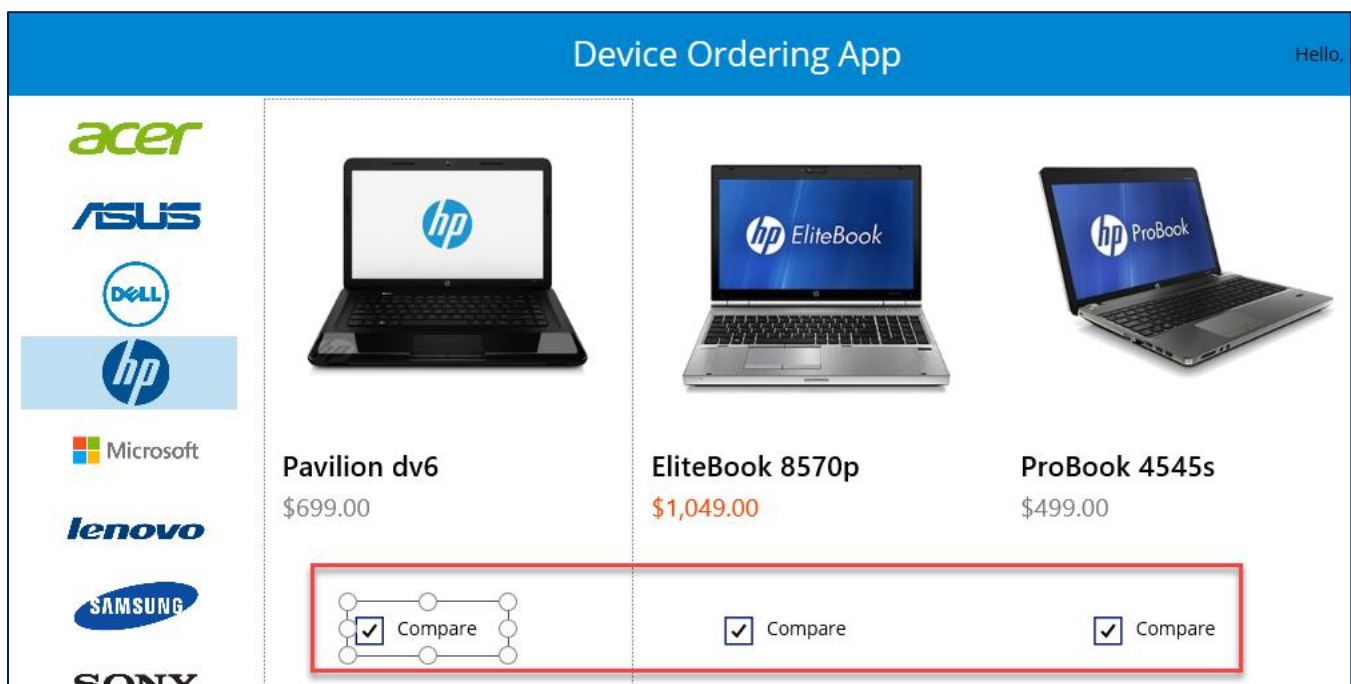


Task 2: Create the edit form

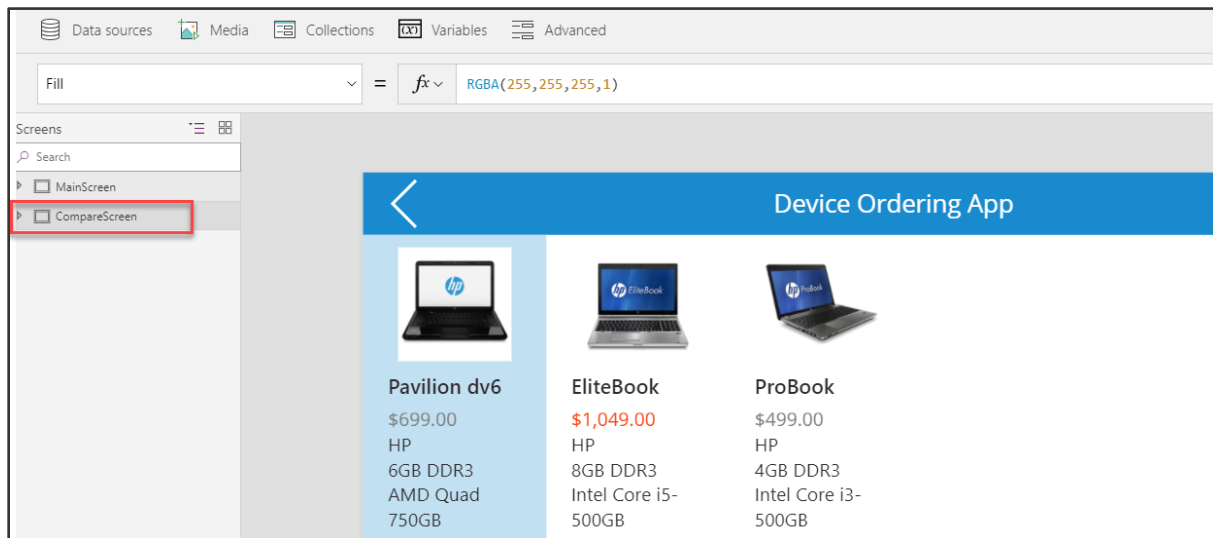
1. Select the **MainScreen**.



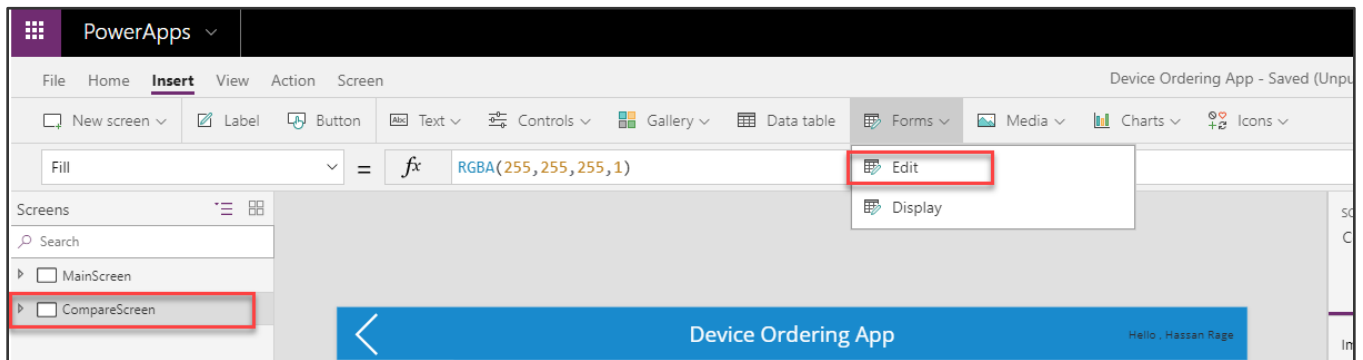
2. Select few devices. Hold the "Alt" key, and then it will allow you to check the compare on the devices



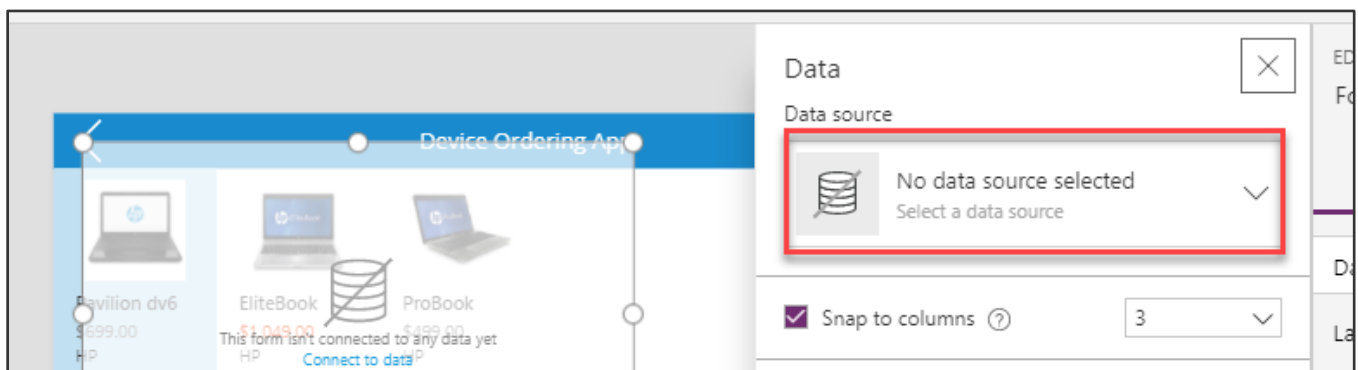
3. Select the **CompareScreen**. You will now have the selected devices.



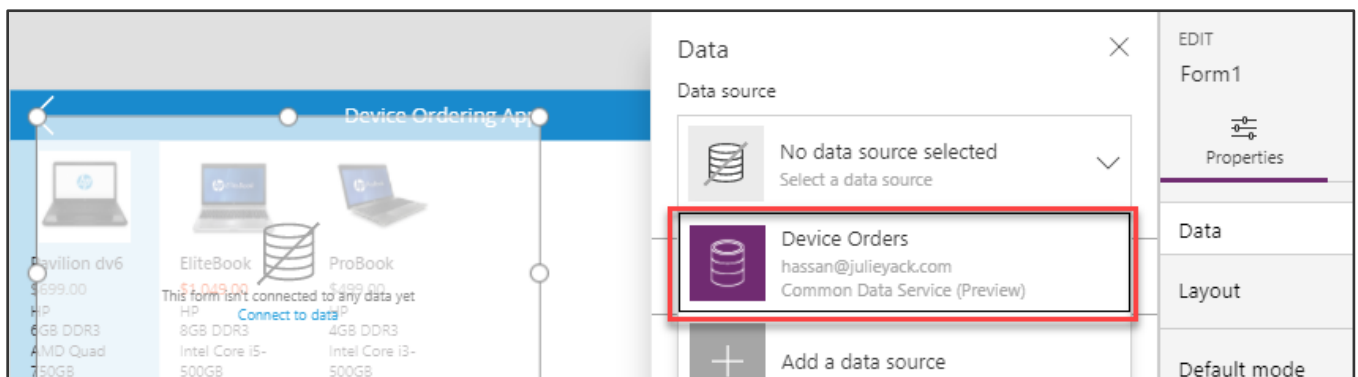
4. Select the **Insert** tab, click **Forms**, and select **Edit**.



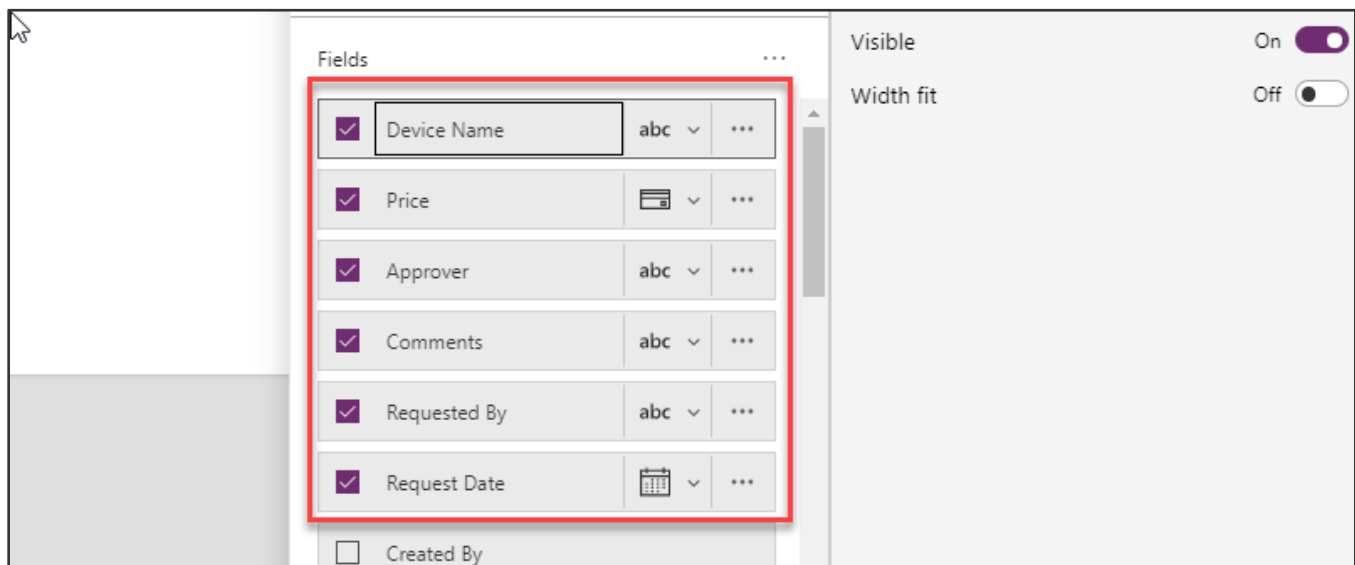
5. Click the **Data source** drop-down in the Data pane on the right.



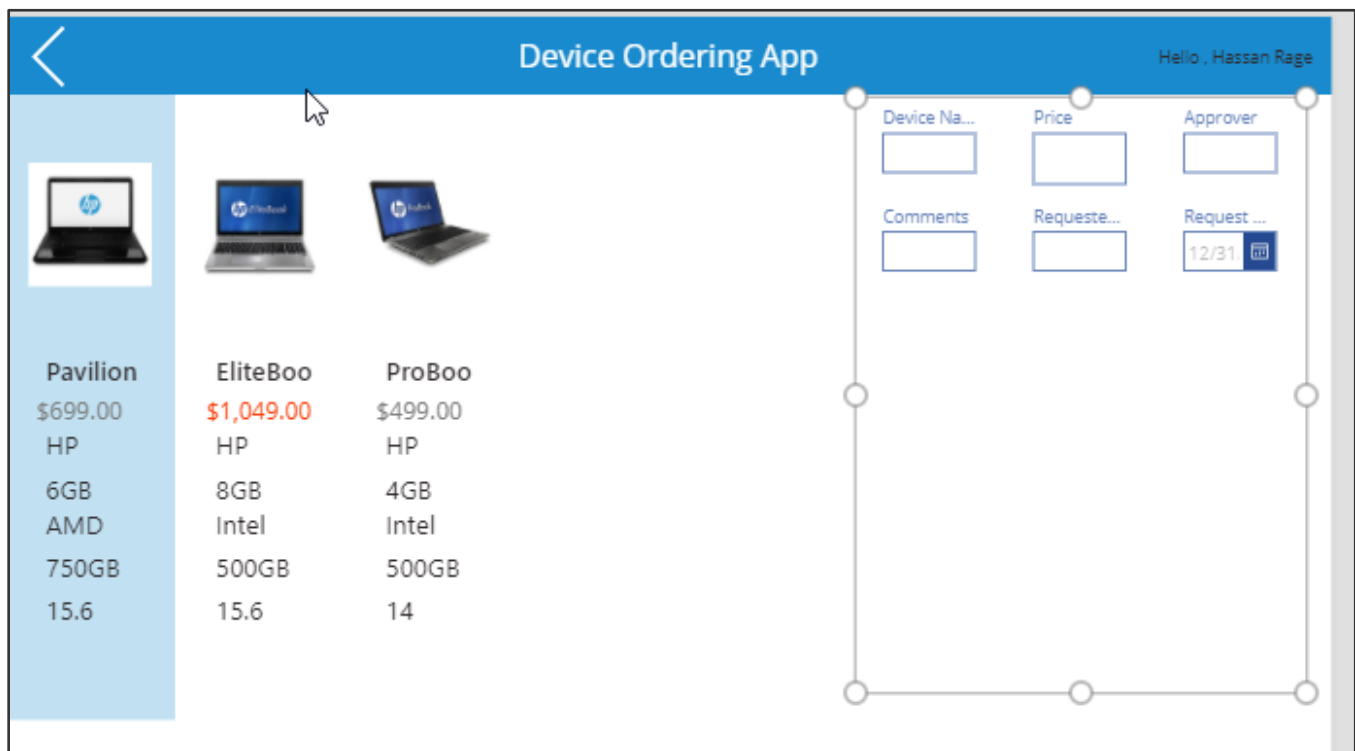
6. Select the **Device Orders** entity as the data source.



7. Select the fields shown in the image then close the Data pane.

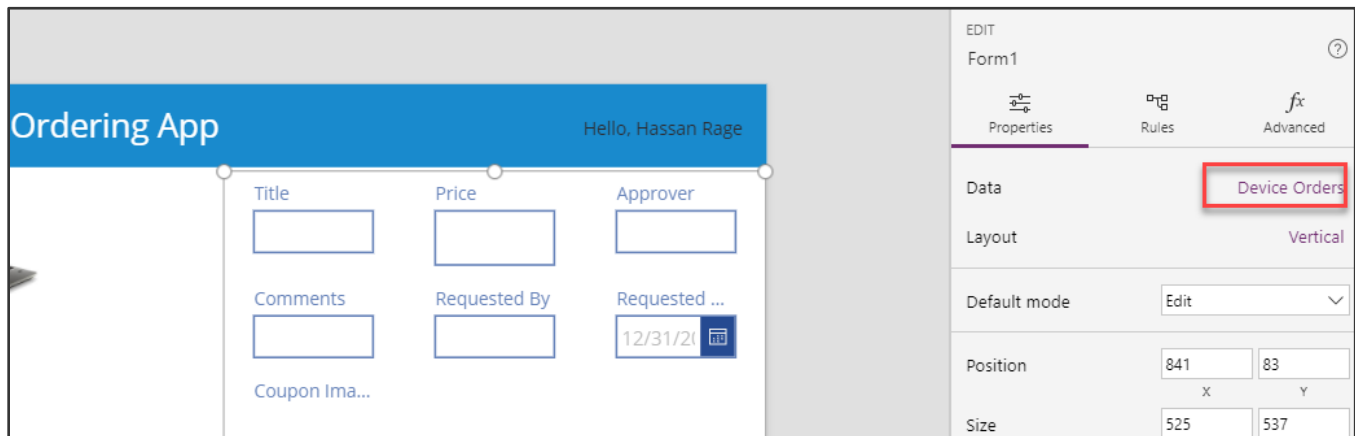


8. Move the form control Form1 to the right of the screen and resize it using the drag handles such that it fits in the empty space. See picture on the right. Make sure there is enough space below the form to add a Submit button.

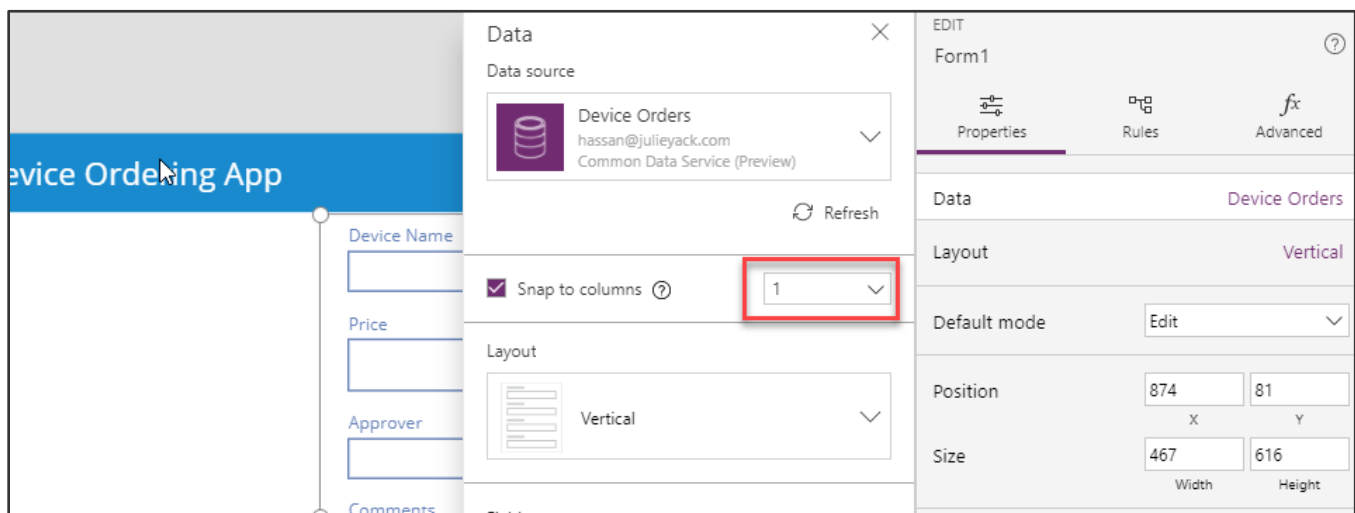


Note: You can always select controls, such as the Form1 control, from the tree view on the left to make sure you are selecting the correct control. To move it make sure you select the Form and not a control within the form.

9. Click the **Data** link on the right to open the Data pane.

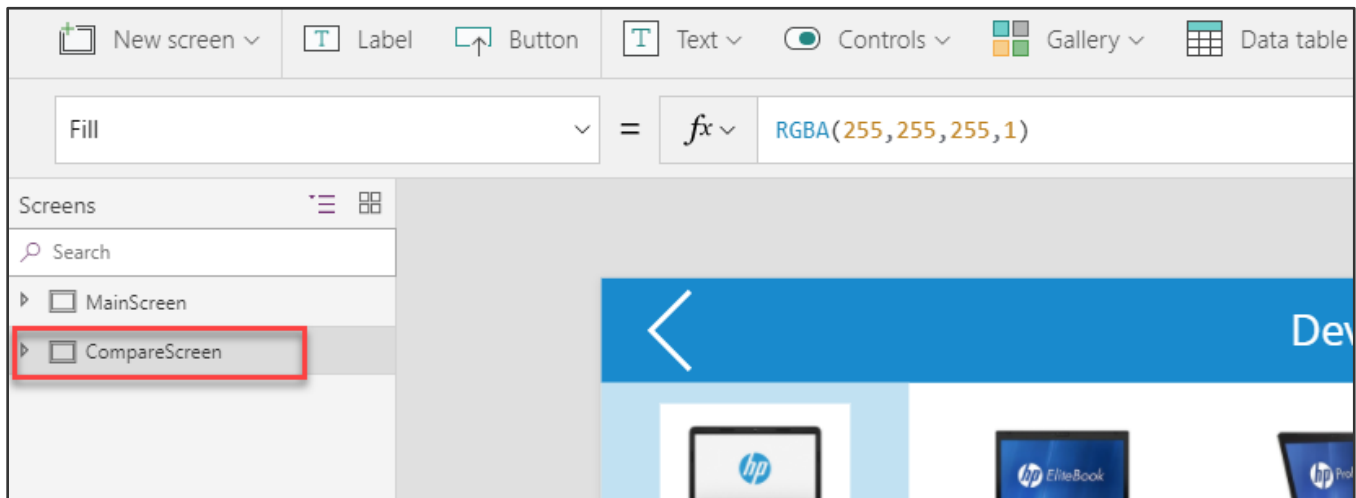


10. Change the **Snap to columns** setting from 3 to 1. This will modify the layout of the edit form to be single column. **Close the Data pane.**

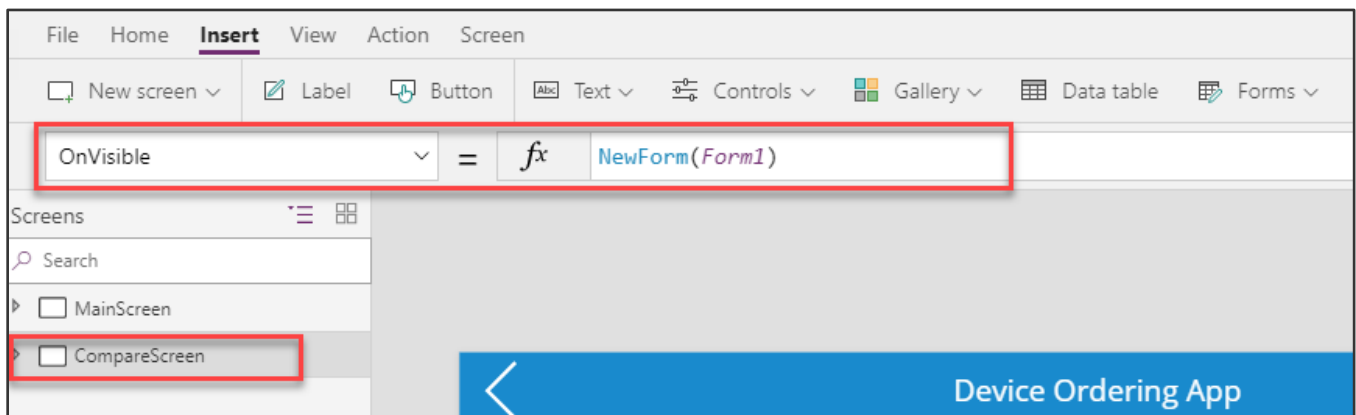


For more info on working with multi-column form layouts, see <https://powerapps.microsoft.com/tutorials/working-with-form-layout/>.

11. To create a new instance of the form when the screen is loaded. Click **CompareScreen** in left tree view pane.



12. Select the **OnVisible** property of the screen, enter: **NewForm(Form1)**.

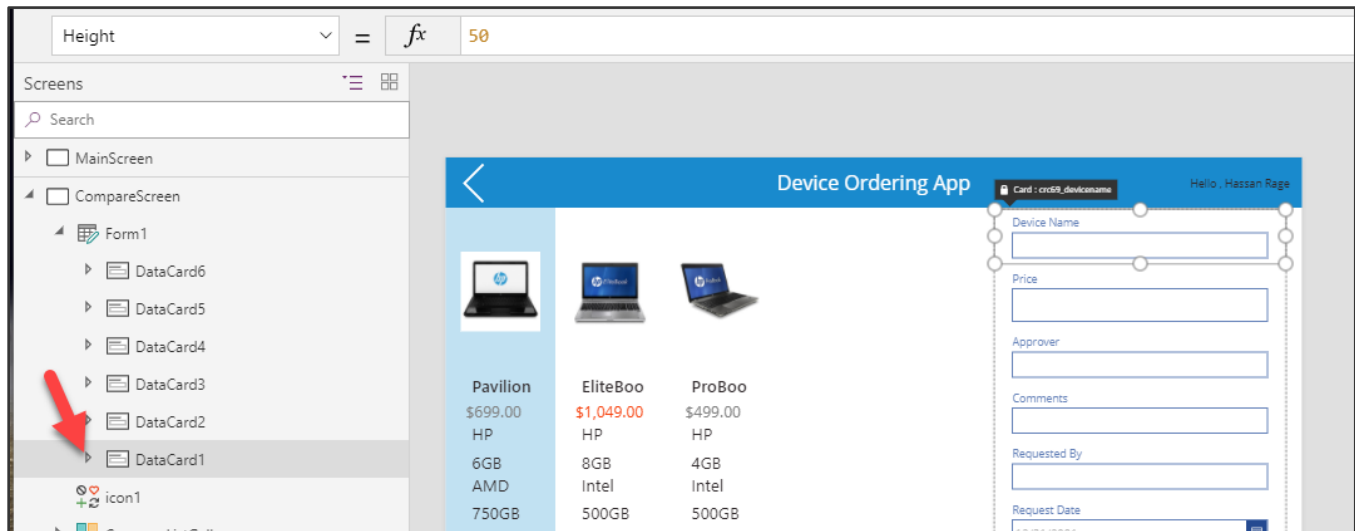


Task 3: Configure the title field

In the next few steps, you will configure each of the form fields.

Let's start by configuring the Title to display the manufacturer and device name for the selected device. For example, if the user selects the Surface Pro device, we want the device order to have the title: "Microsoft – Surface Pro".

1. Expand the **Title card**. The control name is **DataCard1**.



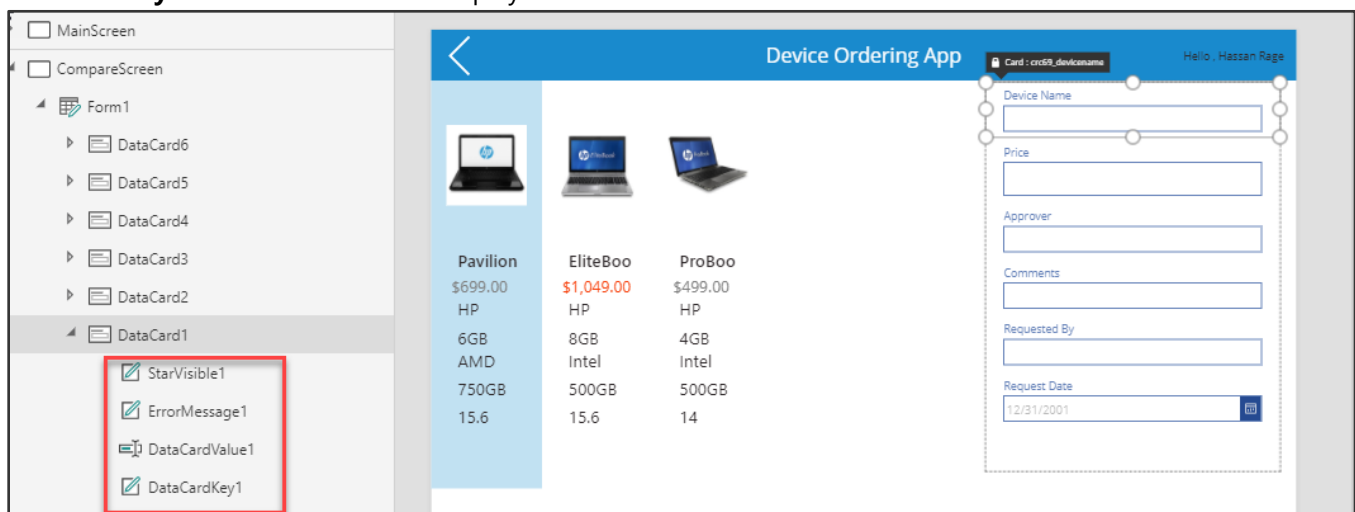
Notice that the default card contains a few controls:

StarVisible1: This is a label control that has an asterisk (*) which has its Visible property set to true or false depending on whether the field is Required or not. Since the Title field was marked as Required when you configured the entity, its Required property is set to true.

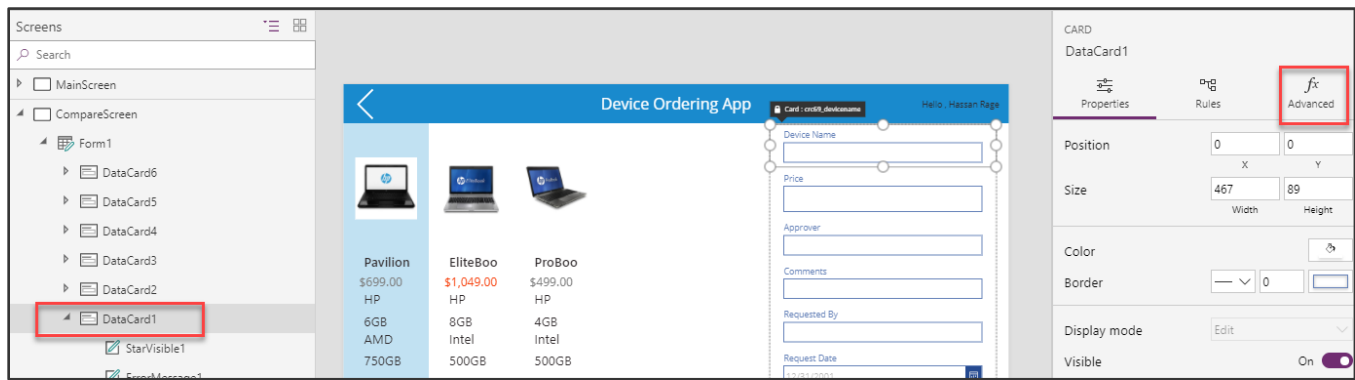
ErrorMessage1: This is a label that is just below the main data entry field which displays error messages.

DataCardValue1: This is the text input control where you can enter the Title. For this scenario, we will set the title based on the selected device.

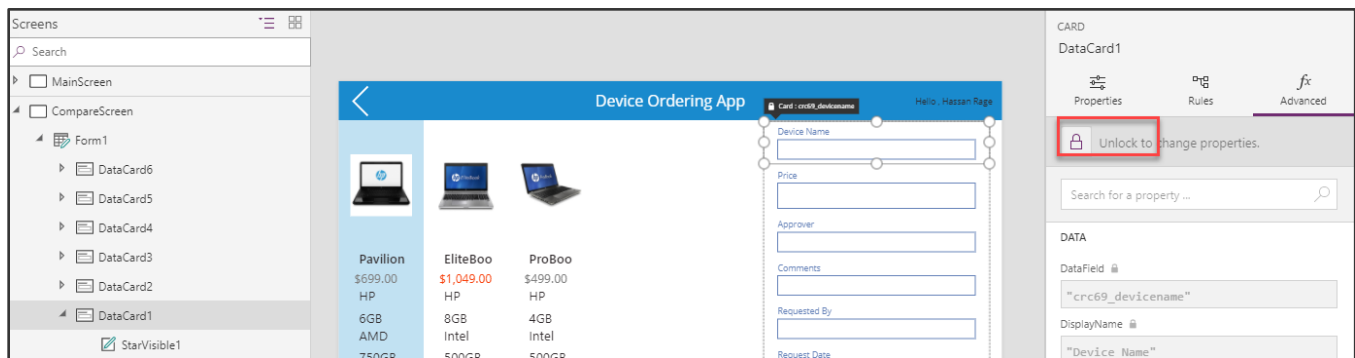
DataCardKey1: This is the label that displays the title of the field.



2. Select **DataCard1** and click **Advanced**.

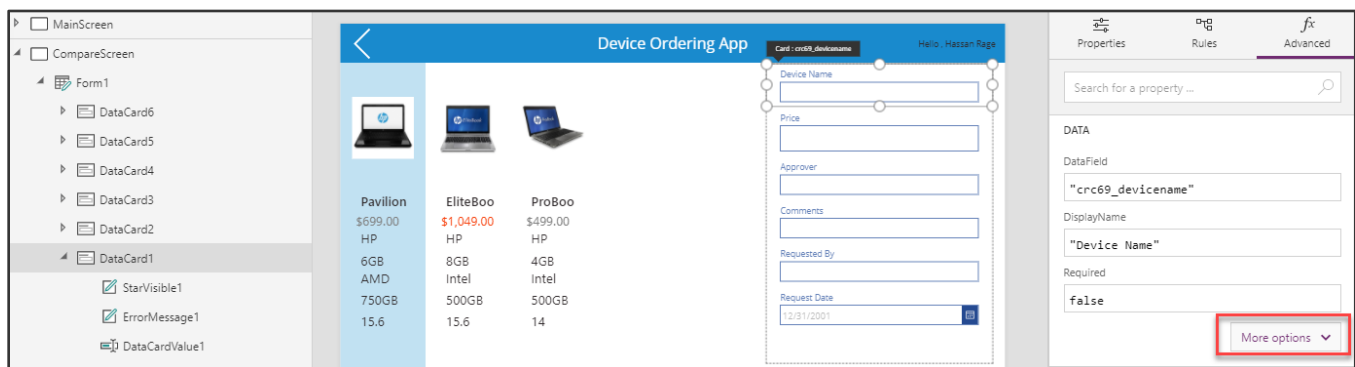


3. Click **Unlock** so you can customize the card



For the next few steps, we will use the Advanced pane to customize control properties within the form, note that you can perform the same customizations using the property drop-down and formula bar in the top left of the studio.

4. Click **More Options** button in the **DATA** section of the Advanced pane.



5. To display the selected item in the Title field, set the **Default** property to `CompareListGallery.Selected.ManufacturerName & " - " & CompareListGallery.Selected.Title`

The screenshot shows the 'Device Ordering App' interface with a list of devices (Pavilion, EliteBoo, ProBoo) and a form for ordering. The 'DataCard1' properties pane is open, showing the 'Default' property set to the formula: `CompareListGallery.Selected.ManufacturerName & " - " & CompareListGallery.Selected.Title`.

6. Click **More Options** button in the **DESIGN** section of the Advanced pane. We are going to change the Device Name field to be read only so they don't change it.

The screenshot shows the 'Device Ordering App' interface with a list of devices (Pavilion, EliteBoo, ProBoo) and a form for ordering. The 'DataCard1' properties pane is open, showing the 'DESIGN' section. The 'More options' button is highlighted in the 'DESIGN' section.

7. Change the **DisplayMode** to **DisplayMode.View**. This will prevent users from changing the value within the text box.

The screenshot shows the 'Device Ordering App' interface. On the left, there's a list of laptop models: Pavilion, EliteBoo, and ProBoo, each with its price, brand, and specifications. In the center, a data card for 'HP - Pavilion dv6' is displayed with fields for Device Name, Price, Approver, Comments, Requested By, and Request Date. On the right, the 'DataCard1' properties pane is open, showing various properties like Width, WidthFit, Height, Visible, and DisplayMode. The 'DisplayMode' property is highlighted with a red box and set to 'DisplayMode.View'.

Model	Price	Brand	RAM	Storage	Screen
Pavilion	\$699.00	HP	6GB	750GB	15.6
EliteBoo	\$1,049.00	HP	8GB	500GB	15.6
ProBoo	\$499.00	HP	4GB	500GB	14

Device Name: HP - Pavilion dv6

Price: [Text Box]

Approver: [Text Box]

Comments: [Text Box]

Requested By: [Text Box]

Request Date: 12/31/2001

Properties | Rules | **Advanced**

Width: 467

WidthFit: false

Height: 50

Visible: true

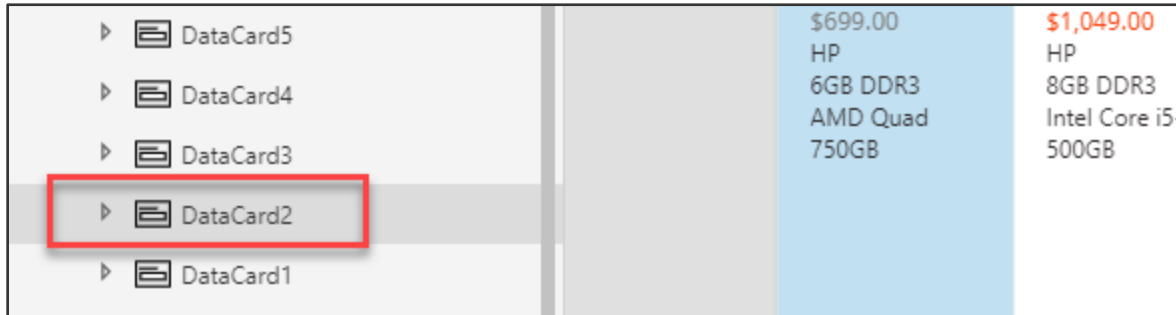
DisplayMode: DisplayMode.View

Fewer options ^

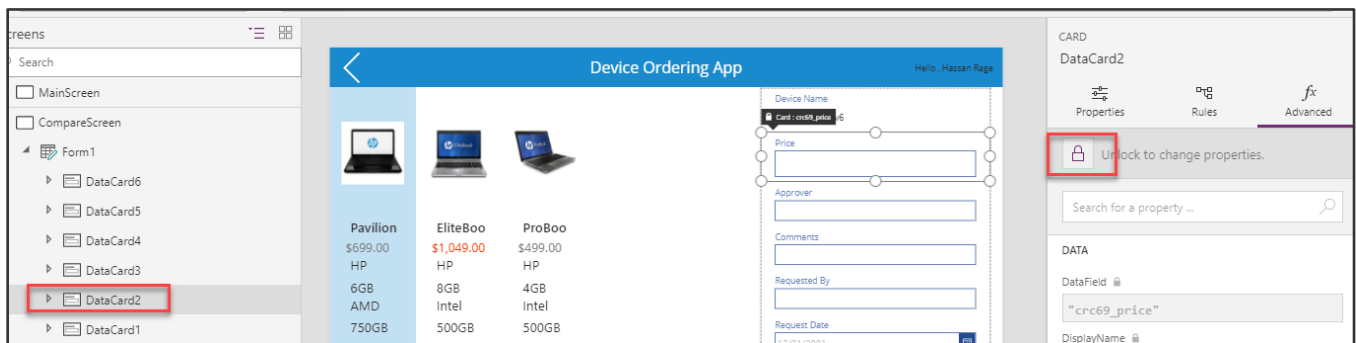
Task 4: Configure the price field

In this task, we are going to set the price to the price of the item and then make it read-only.

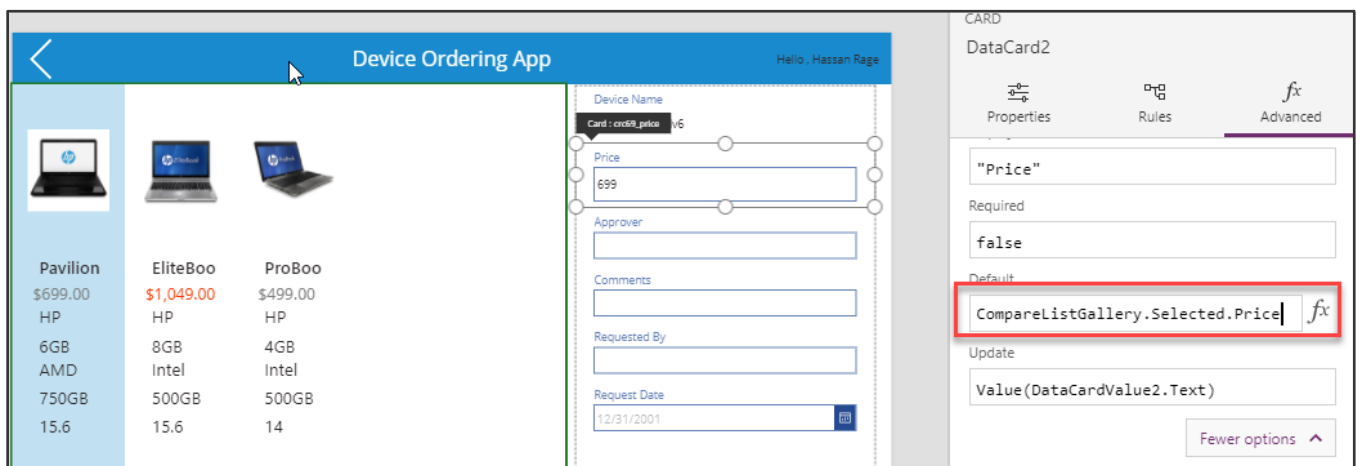
1. Select DataCard2.



2. Select the **Advanced** tab and click **Unlock**.



3. Change the **Default** property to: `CompareListGallery.Selected.Price`



4. Change the **DisplayMode** property to: `DisplayMode.View`

The screenshot shows the Power Platform App Designer interface. On the left, a card view displays a laptop icon and the following text: "ProBoo", "\$499.00", "HP", "4GB", "Intel", "500GB", and "14". On the right, the Properties pane is open, showing the "Advanced" tab. The "DisplayMode" property is highlighted with a red box and set to "DisplayMode.View".

Properties pane (Advanced tab):

- Width: 467
- WidthFit: false
- Height: 50
- Visible: true
- DisplayMode: **DisplayMode.View**

Other properties visible in the Properties pane:

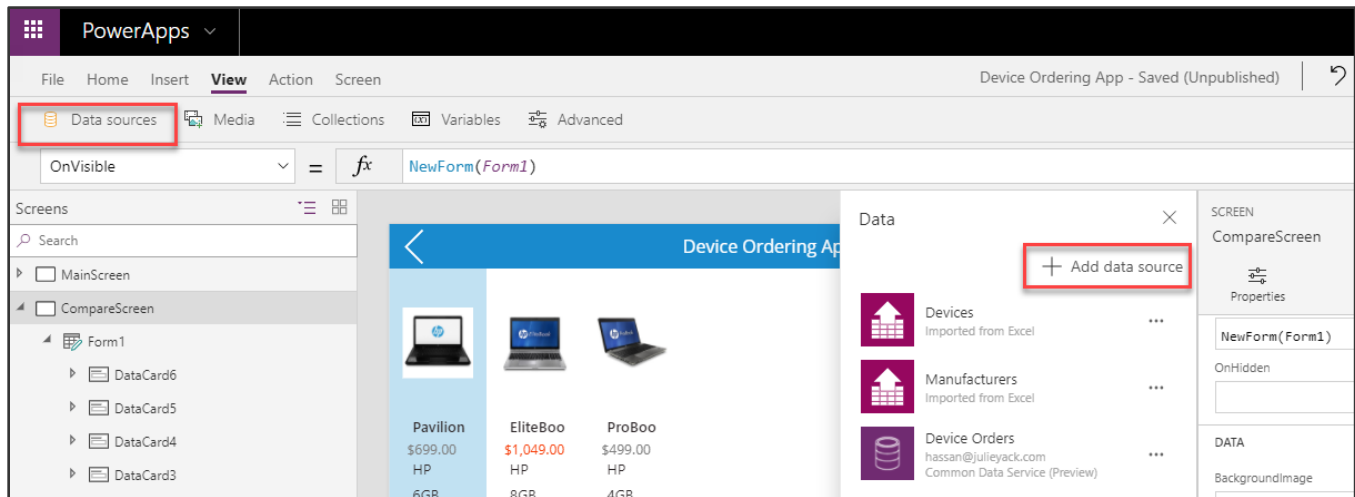
- Device Name: Card : cre59_price
- Price: 699
- Approver: [Text Input]
- Comments: [Text Input]
- Requested By: [Text Input]
- Request Date: 12/31/2001

Task 5: Configure the approval field

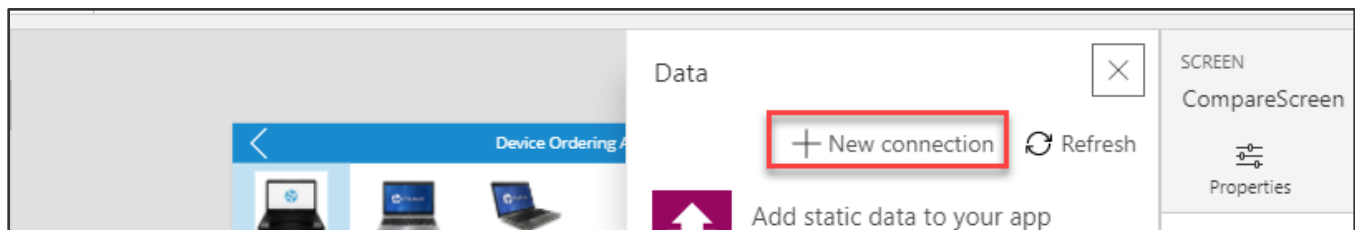
Let's set the **default** value for the Approver to be the email address of the **logged in user's manager**.

You will use the **Office 365 graph** to retrieve the manager's email. You can find more about the Office 365 Users Connection provider here <https://docs.microsoft.com/connectors/office365users/>

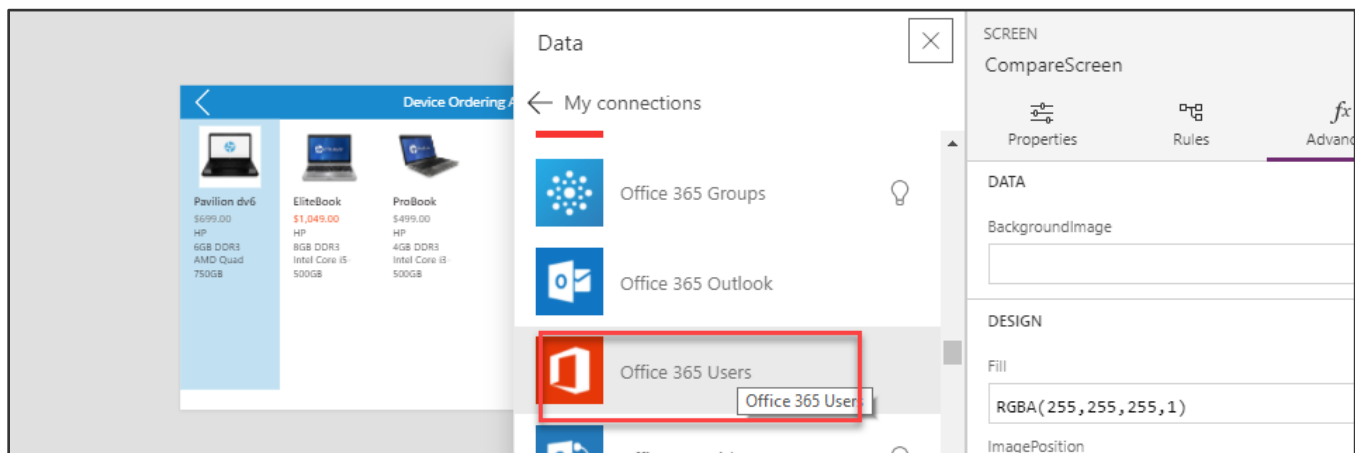
1. Select **View** in the ribbon, then **Data sources**. In the right-hand Data pane, select **+Add data source**.

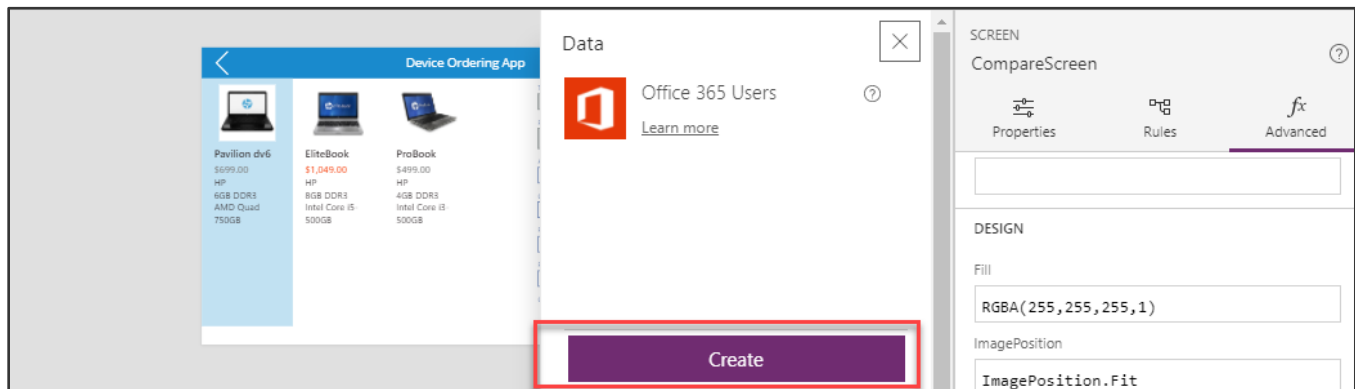
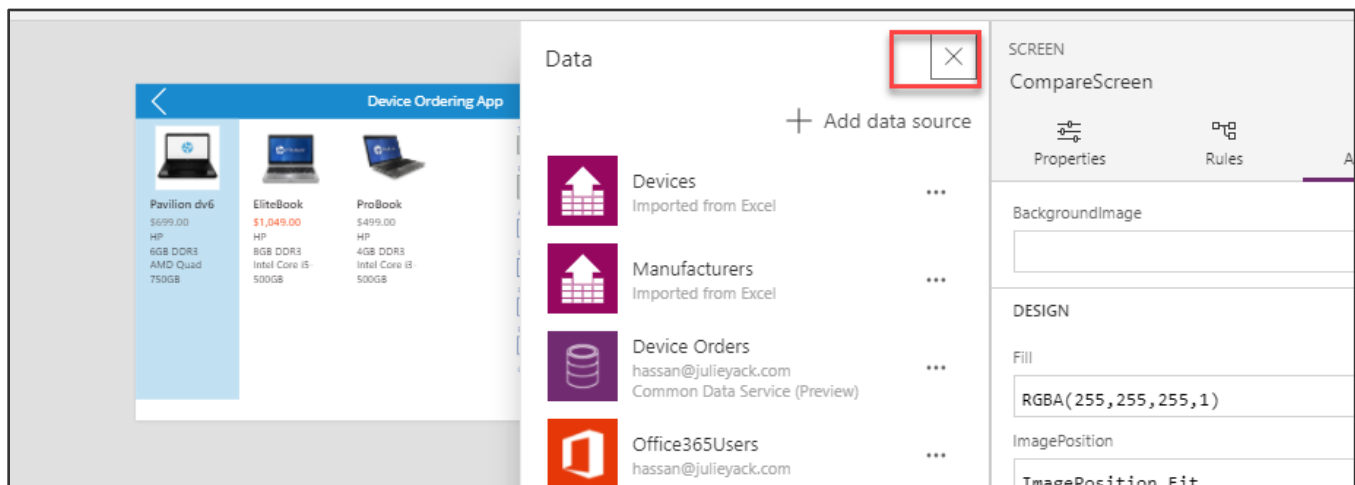
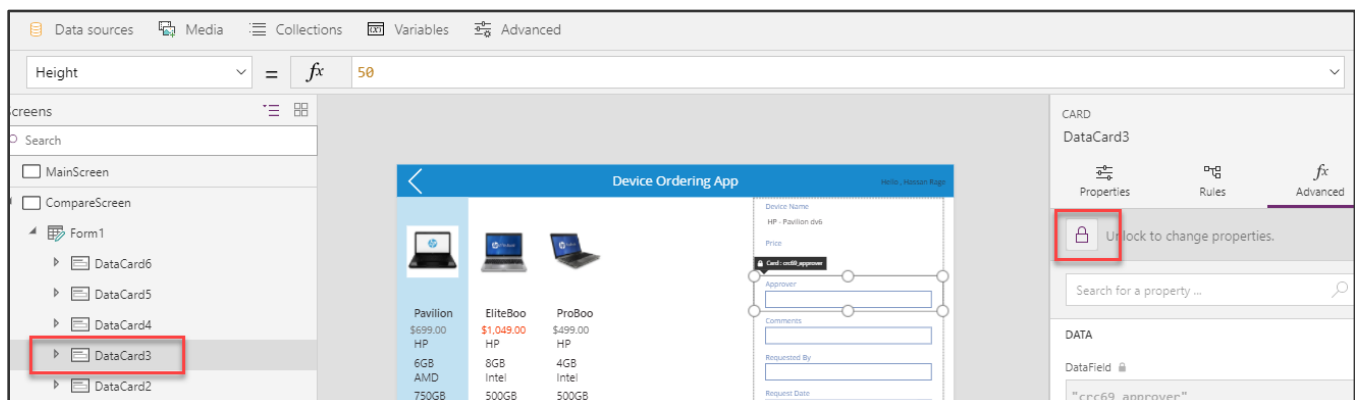


2. Click **+New connection**.

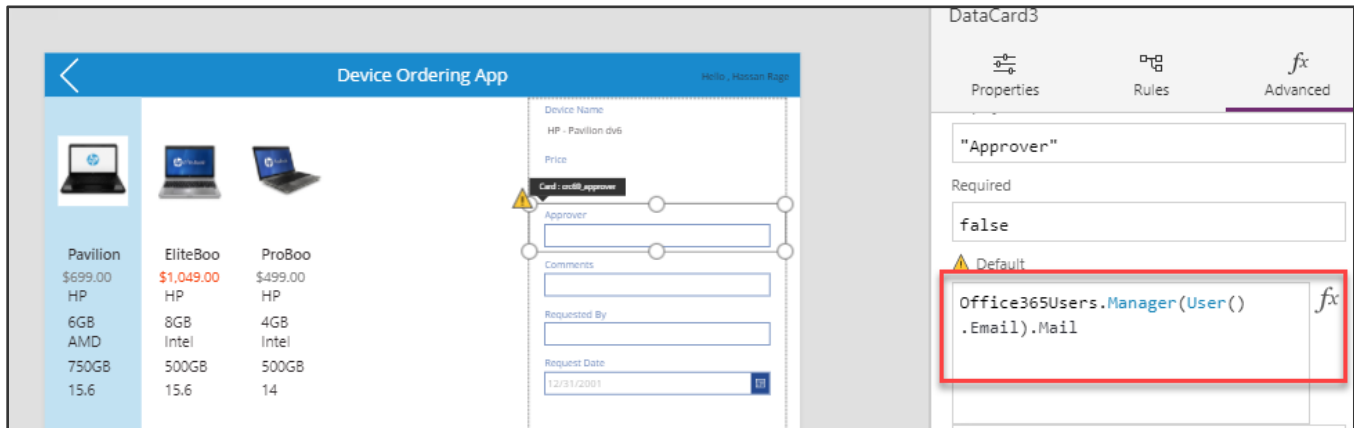


3. Select the connection type - **Office 365 Users**.



4. Click **Create**.5. Close the **Data** pane.6. Select the **Approver** card also known as **DataCard3**.7. Go to the **Advanced** pane and **Unlock**.

8. Set the Default value to: `Office365Users.Manager(User()).Email).Mail`



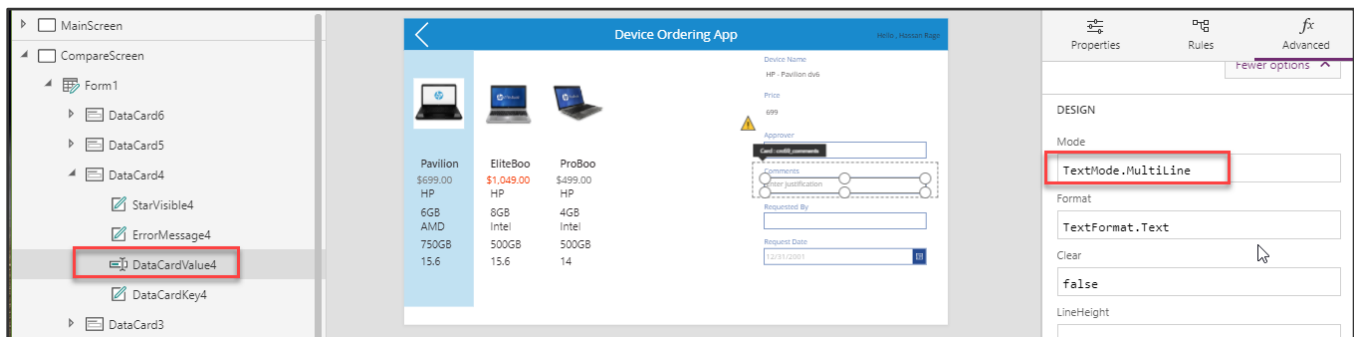
Note: If you hit an error when calling the `Office365Users.Manager()` function, this may be because a manager is not set up in the system for the logged in Office 365 user. In that case, you can skip this step and simply hard code the approver's email.

This will make an API call at runtime to get the manager's email address of the logged-on user.

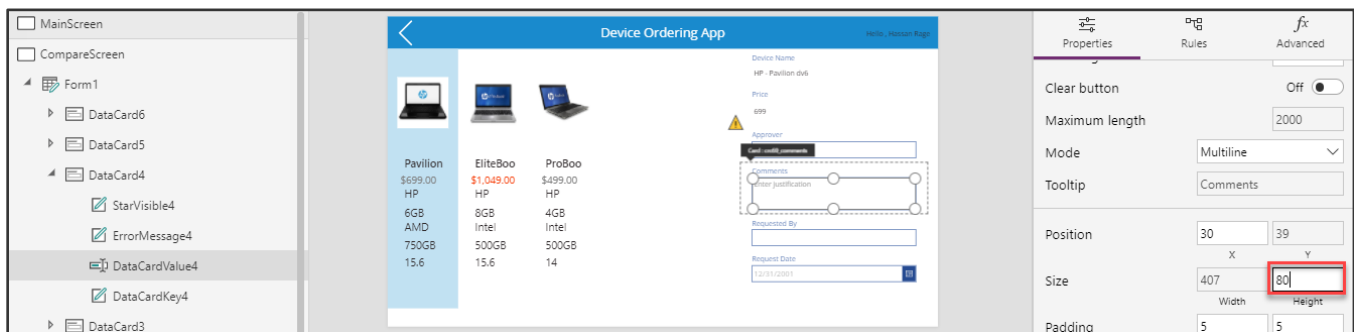
In a subsequent step, you will create a Flow that sends an Approval request to the specified Approver. For test purposes, it's a good idea to enter your own email address in the Approver text box when you run the application.

Task 6: Configure the Comment field

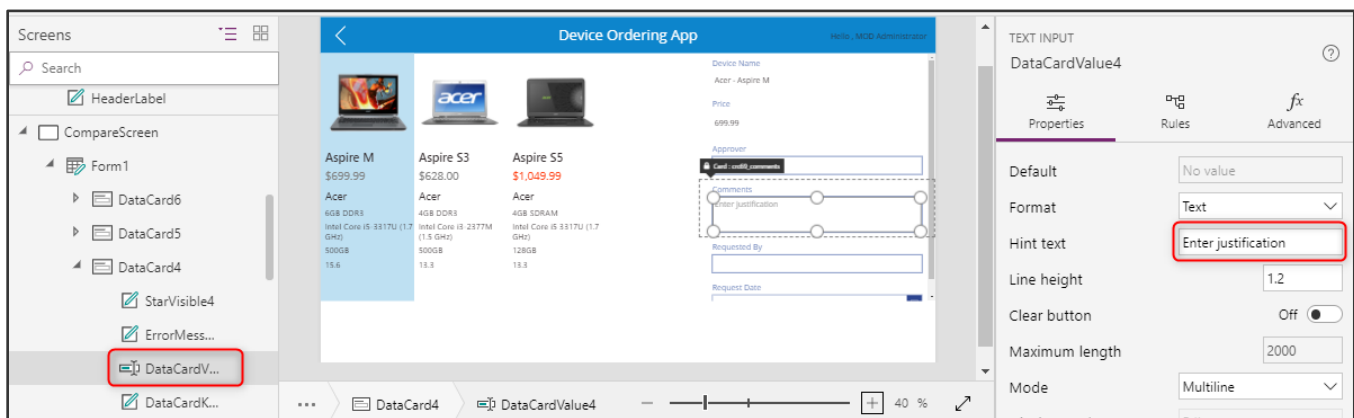
1. Select **DataCard4** -> **DataCardValue4** and change the **Mode** to **TextMode.Multiline**



2. Select the **Properties** tab and change the **Size -> Height** value to **80**.



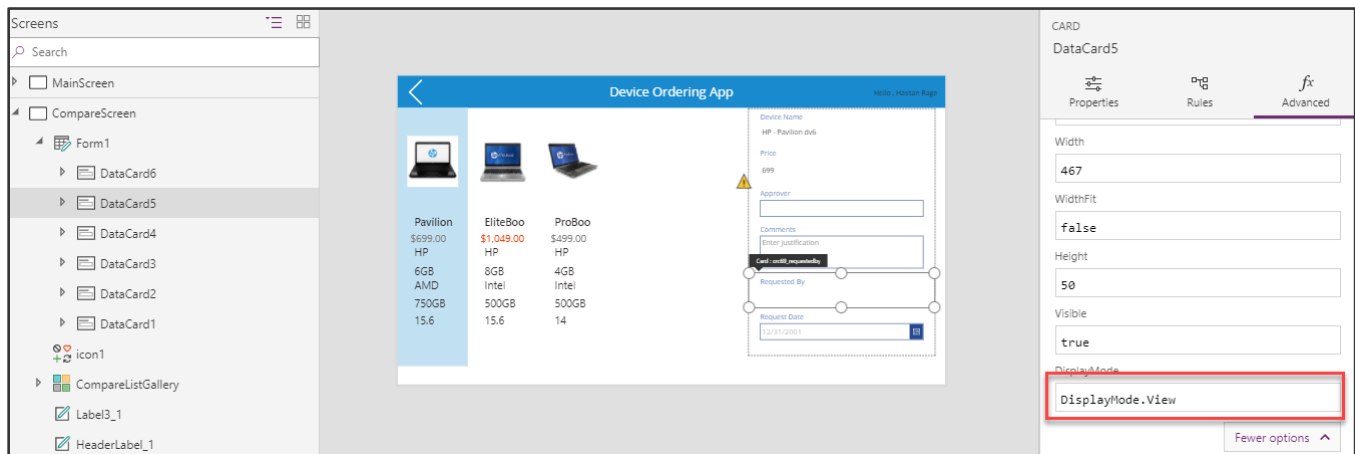
Optionally, you may select the Text Input control **DataCardValue4** and set its **HintText** property to: "Enter justification" (without quotes).



Task 7: Configure the Requested By field

Let's set the Requested By field to be the current logged on user's email, and disable the control so the user cannot change this value.

1. Select the Requested By card, also known as **DataCard5**
2. Go to the **Advanced** pane and **Unlock** the card.
3. Change the **DisplayMode** property to: `DisplayMode.View`



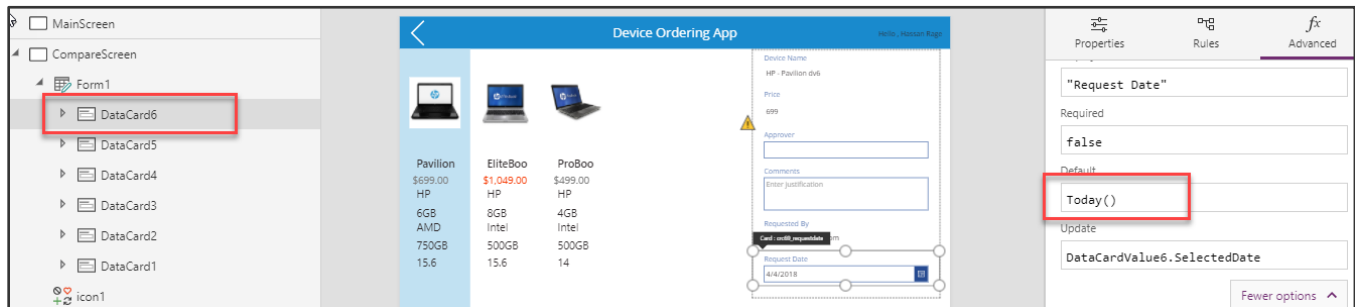
4. Set the **Default** value to `User().Email`
This is the email of the currently logged in user



Task 8: Configure the requested date field

Let's set the Request Date to be today's date.

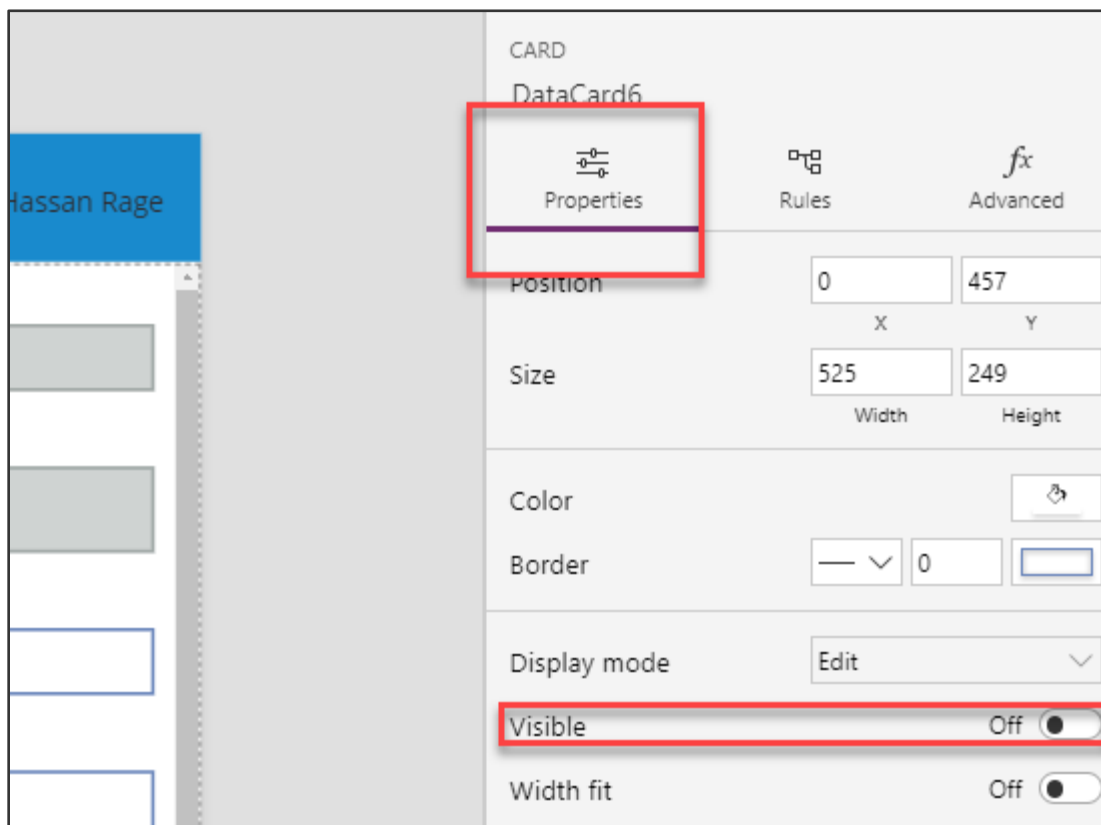
1. Select the Request Date card, also known as **DataCard6**.
2. Go to the **Advanced** pane and **Unlock the card**.
3. Change the **Default** property to `Today()`



Notice that the date in the calendar control will change to today's date.

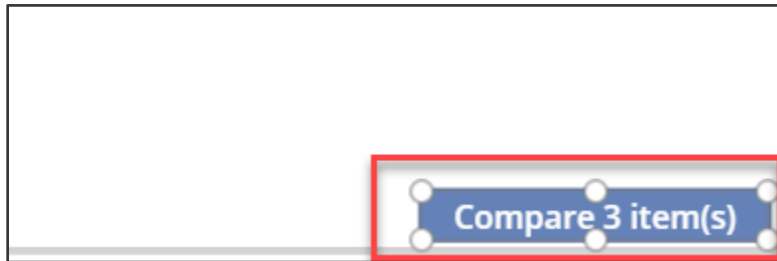
Now we will hide the Request Date card. We don't need to show this field to the user. Since we have included it as part of the form the field will get updated as part of the form submit.

4. Go to the **Properties** pane on the right.
5. Toggle the **Visible** property to **Off**.



Task 9: Add a button to submit the form

1. Select the **MainScreen**.
2. Copy the **Compare button** from the first screen which has the correct color values.



3. Go back to the **CompareScreen** and paste (Ctrl-V) the button.

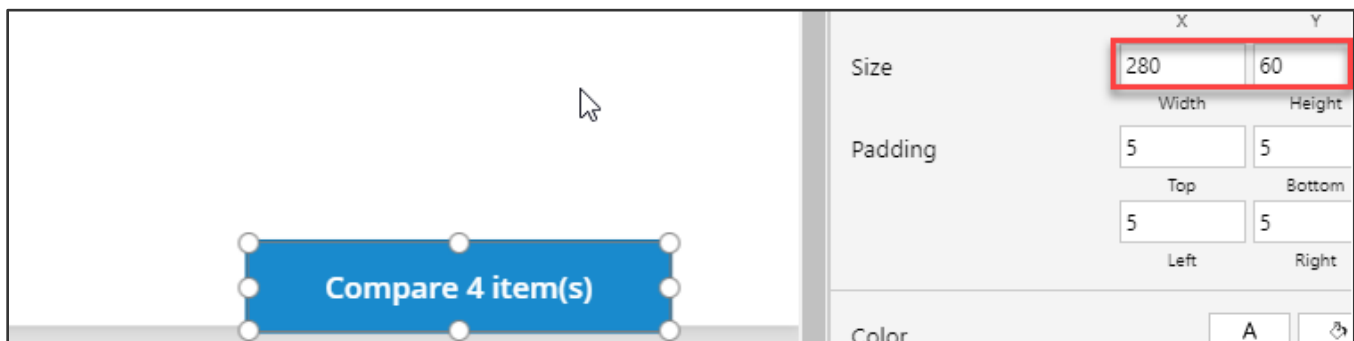
teBoo	ProBoo	
049.00	\$499.00	
	HP	
GB	4GB	
tel	Intel	
0GB	500GB	
.6	14	

Comments
Enter justification

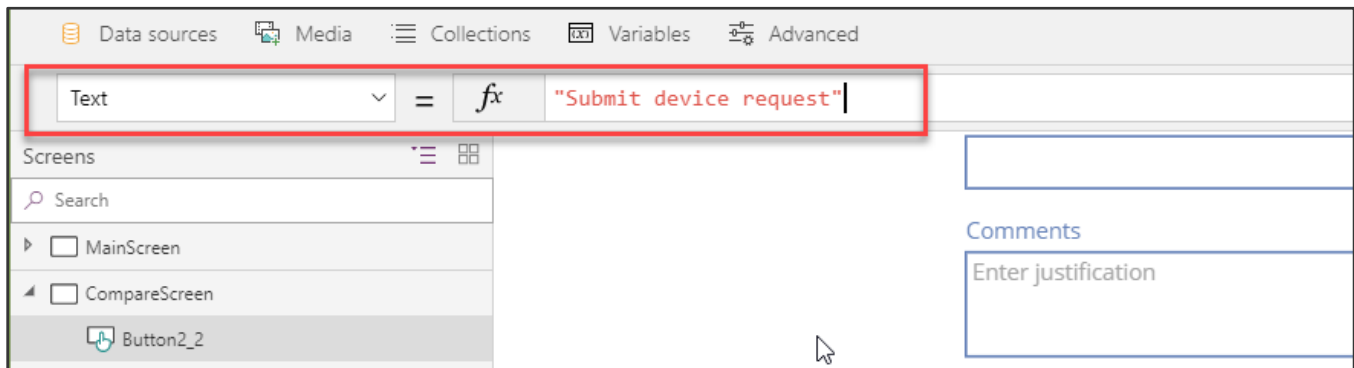
Requested By
hassan@julieyack.com

Compare 4 item(s)

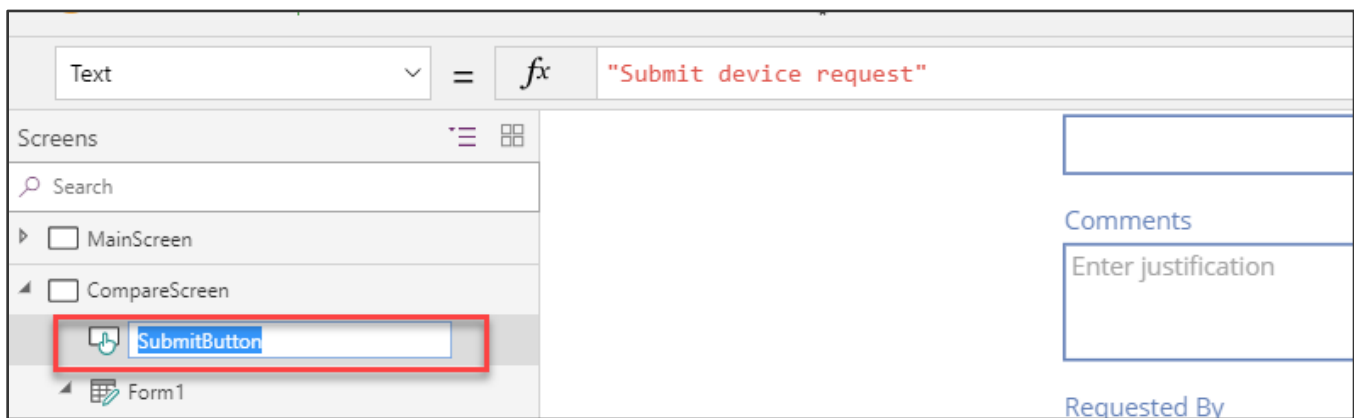
4. Position it in the bottom right of the screen, center aligned with the Form.
5. Make the button larger – you can resize to 280x60 using the Properties pane on the right.



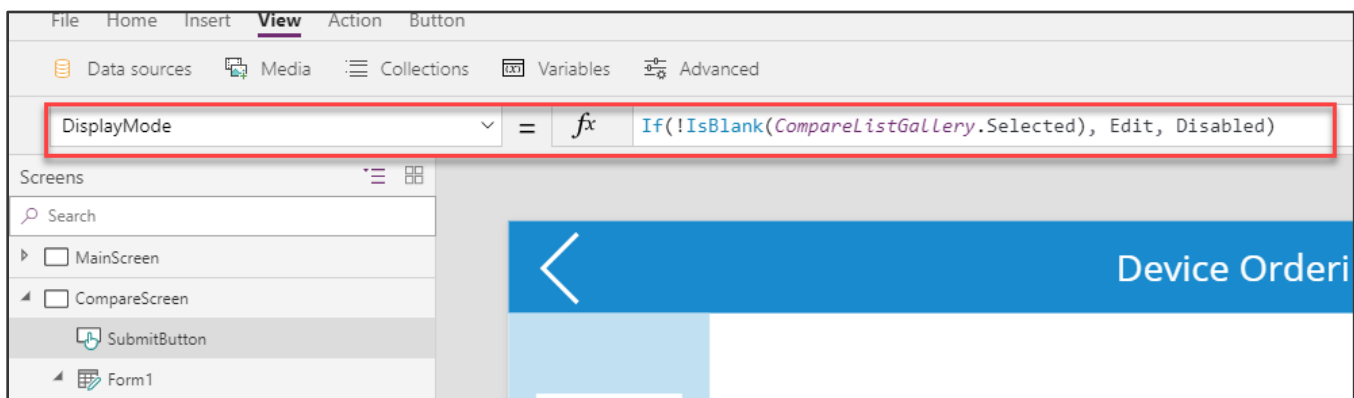
6. Set the button's **Text** property to **"Submit device request"**



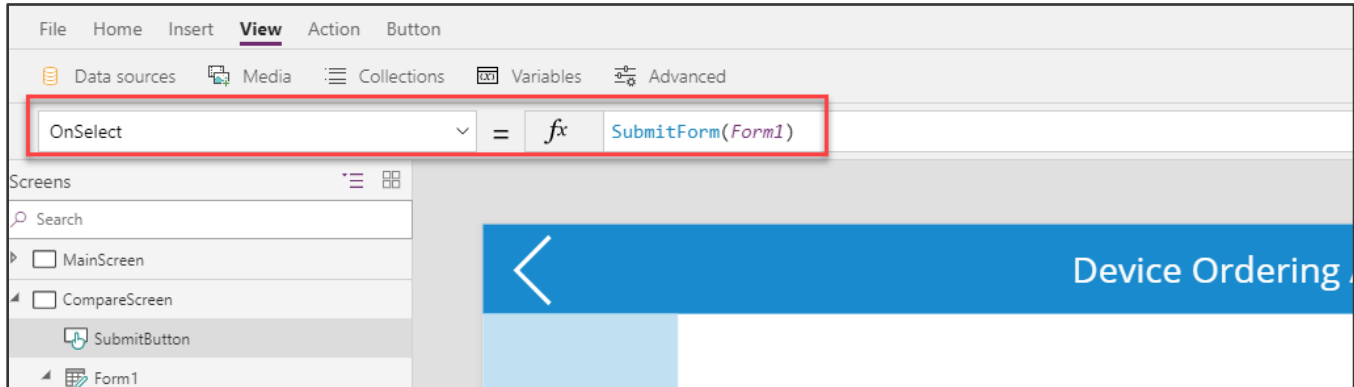
7. Rename the button to **SubmitButton**.



8. The button should be enabled only if a device is selected. To do this, change the button's **DisplayMode** property to: `If(!IsBlank(CompareListGallery.Selected), DisplayMode.Edit, DisplayMode.Disabled)`



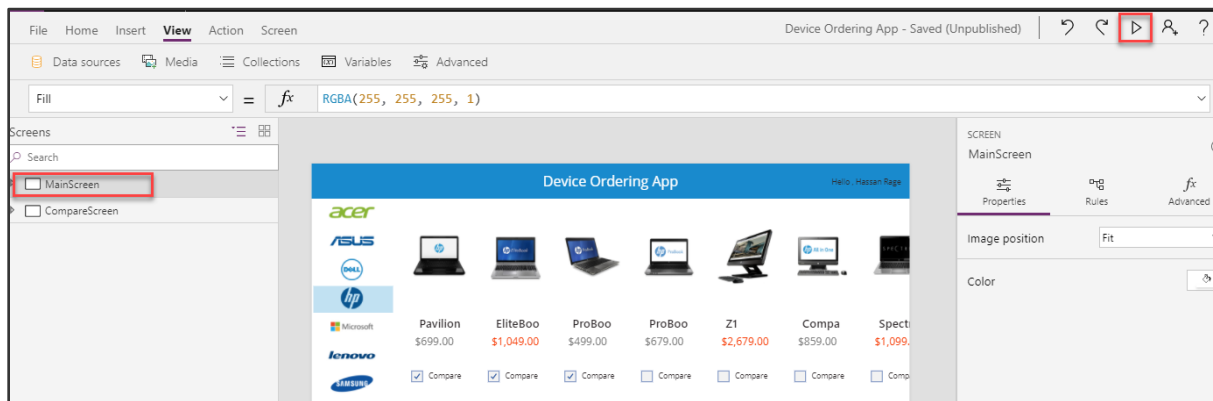
9. Next, we are going to configure what we want to happen when the button is clicked. Set the **OnSelect** property to `SubmitForm(Form1)`



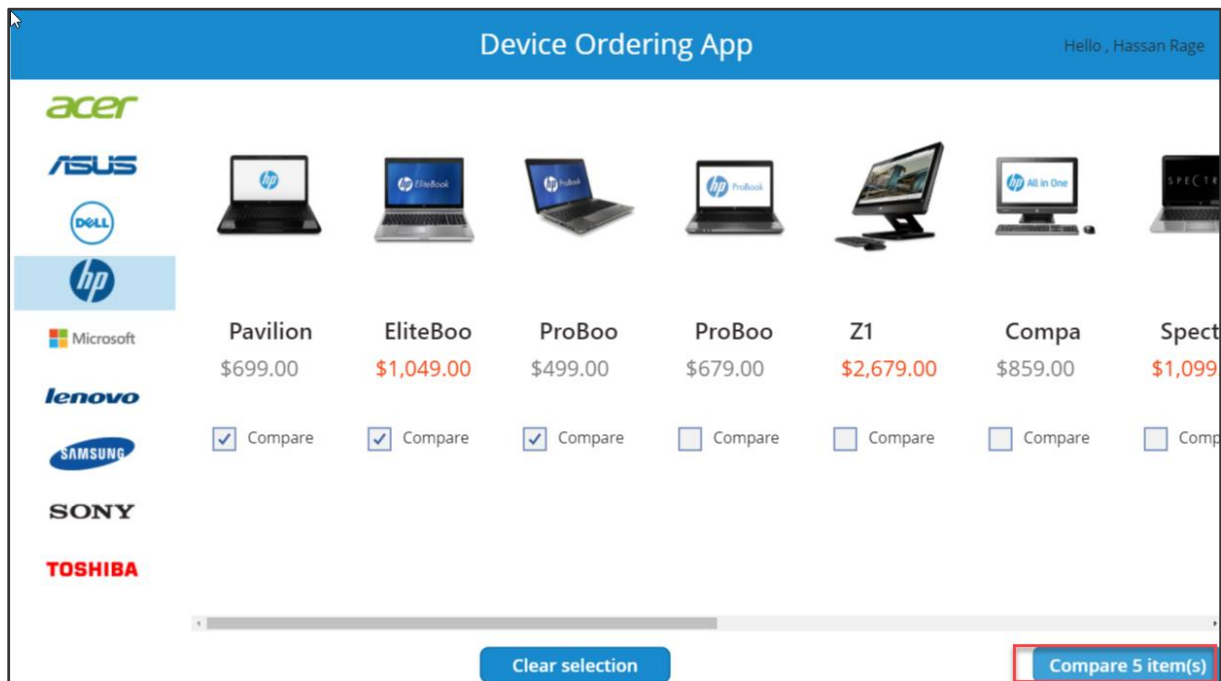
When the button is pressed, the form data will be submitted to the Common Data Service.

Task 10: Test the form

1. Select the **MainScreen** in the left side tree navigation and click Play.



2. Select a few devices to compare. And click **Compare**.



3. Select one of the devices.

The screenshot shows the 'Device Ordering App' interface. On the left, there are three laptop images with their specifications and prices. The 'ProBook' is highlighted with a blue background. On the right, the form fields are partially filled.

Pavilion	EliteBoo	ProBoo
\$699.00	\$1,049.00	\$499.00
HP	HP	HP
6GB	8GB	4GB
AMD	Intel	Intel
750GB	500GB	500GB
15.6	15.6	14

Device Name: HP - ProBook 4545s
Price: 499
Approver:
Comments:
Requested By: hassan@julieyack.com

Submit device request

Notice that the Title, Price and Requested By fields are already filled in.

4. Change the Approver email to your own email for test purposes
5. Add some **Comments**, such as: "Current laptop does not work, need a new device."
6. Click **Submit device request**.

The screenshot shows the 'Device Ordering App' interface with the form fields updated. The 'Approver' field now contains 'hassan@julieyack.com' and the 'Comments' field contains 'Current laptop does not work, I need a new device.' The 'Submit device request' button is highlighted with a red box.

Pavilion	EliteBoo	ProBoo
\$699.00	\$1,049.00	\$499.00
HP	HP	HP
6GB	8GB	4GB
AMD	Intel	Intel
750GB	500GB	500GB
15.6	15.6	14

Device Name: HP - ProBook 4545s
Price: 499
Approver: hassan@julieyack.com
Comments: Current laptop does not work, I need a new device.
Requested By: hassan@julieyack.com

Submit device request

The button should turn disabled (gray) for a few seconds while it's submitting the request. If it does not do this there is likely an error. Click the X in top right to get back to the design mode.

If there is an error, you will see a yellow error icon next to the Submit button, hover over it to check the error.

Pavilion	EliteBook	ProBook
\$699.00	\$1,049.00	\$499.00
HP	HP	HP
6GB	8GB	4GB
AMD	Intel	Intel
750GB	500GB	500GB
15.6	15.6	14

Device Name
HP - ProBook 4545s

Price
499

Approver
hassan@julieyack.com

Comments
Current laptop does not work, I need a new device.

Requested By
hassan@julieyack.com

Submit device request

- Exit the preview mode ('X' in top right)
- Save the Application and Publish

Save

Device Ordering App

Environment: DyAppInDay (org153f370d)

Saved: 9/11/2018 7:31:50 AM

✓ All changes are saved.

Publish See all versions

Publish

Publishing will make this version available to everyone who has Can use permissions.
[Learn more about publishing apps](#)

Device Ordering App

Description
no description

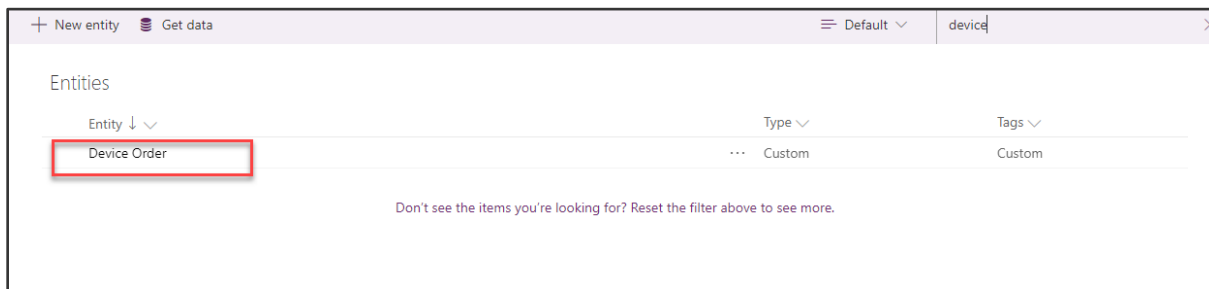
Publish this version Edit details

Task 11: Verify a new item was added to the Device Order entity

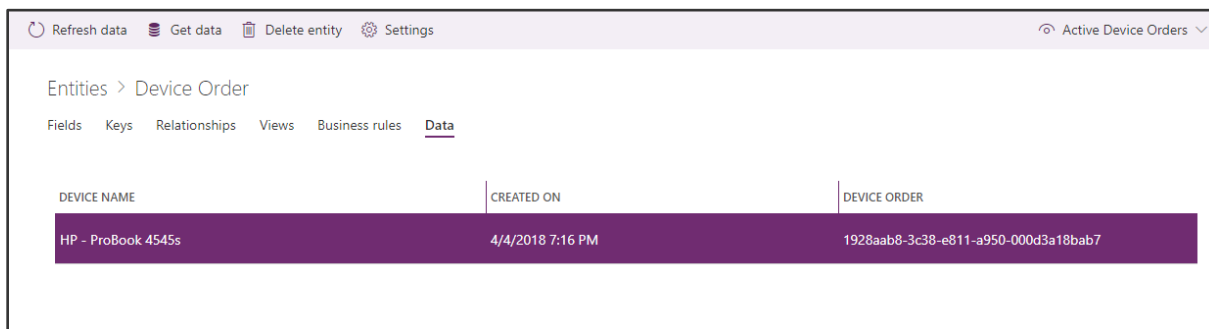
1. Browse a browser window, go to <http://web.powerapps.com>.
2. Click on **Data** -> **Entities**.



3. Select the **Device Order** entity.



4. Select the Data tab.
5. You should see a newly added row with your device order details. This may take a few seconds to load.

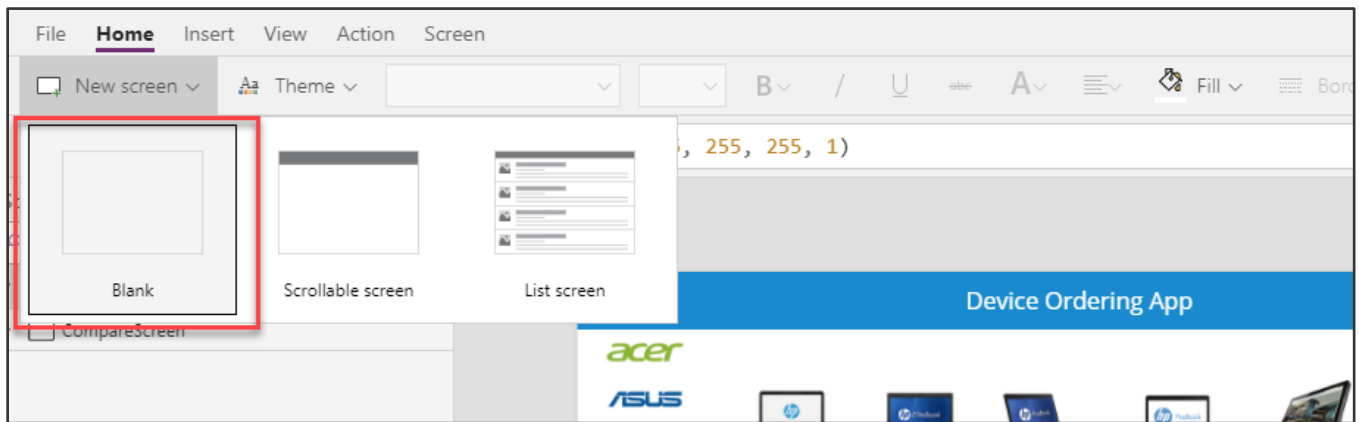


Task 12: [Optional] Navigate to confirmation screen after the Form submit is successful

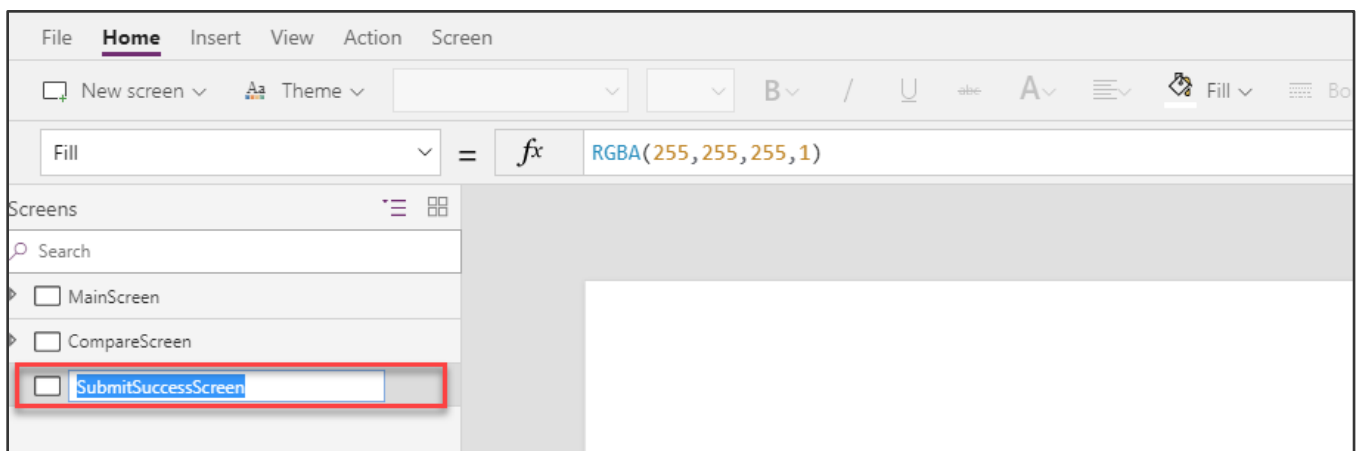
This step is optional, if you're short on time you may skip it and continue on to the next module.

Once the Form has been successfully submitted, it's a good idea to show a confirmation screen and allow the user to navigate back to the main screen.

1. Select **Home** -> **New screen** -> **Blank**

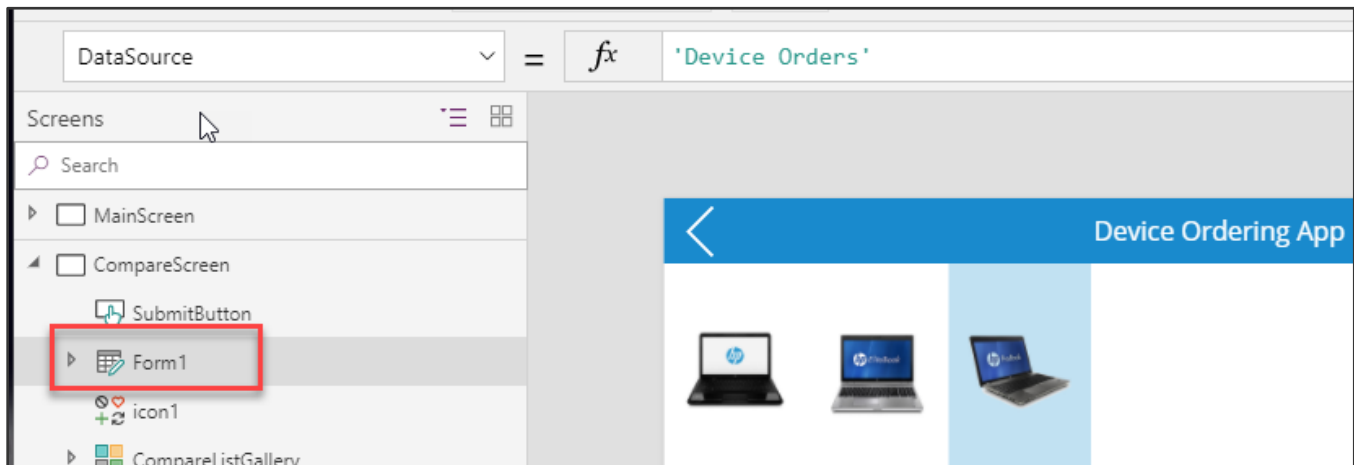


2. Rename the screen to **SubmitSuccessScreen**

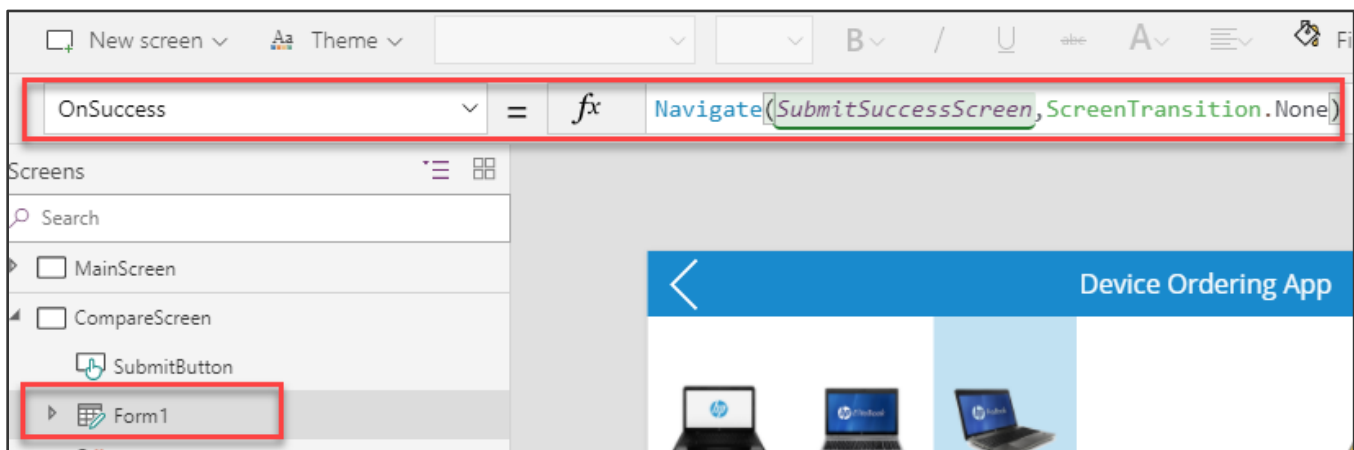


3. Select the **CompareScreen**.

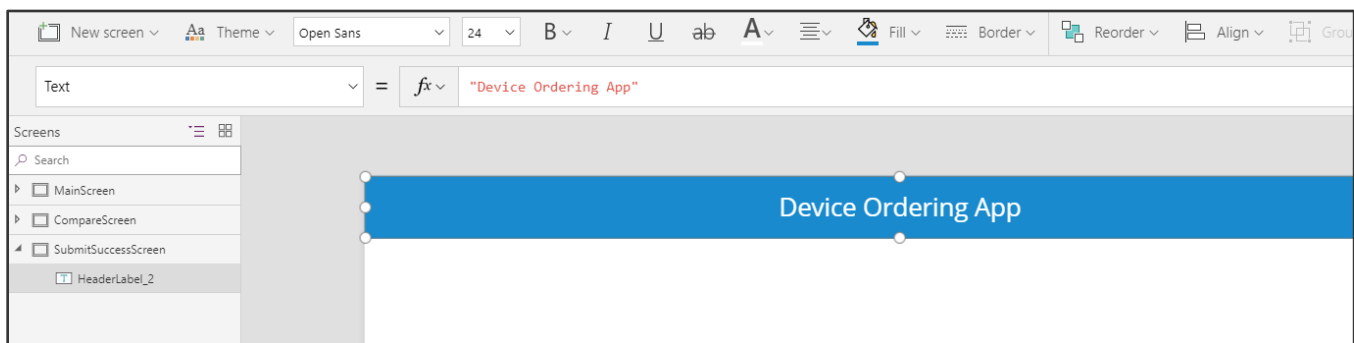
4. Select the Form – you can use the tree view on the left to select **Form1**.



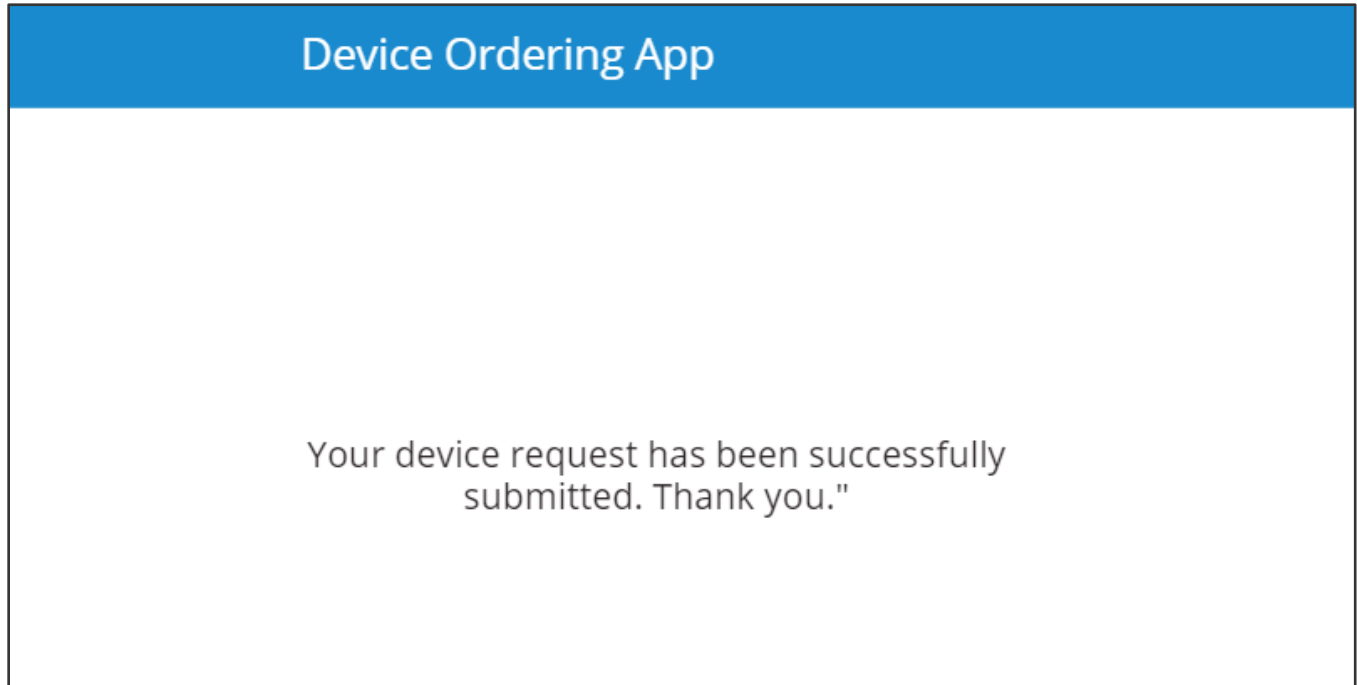
6. Set the **OnSuccess** property to: `Navigate(SubmitSuccessScreen,ScreenTransition.None)`



7. Copy (Ctrl-C) the **header label** from the second screen.
8. Go to the to the **SubmitSuccessScreen** and paste the header label.



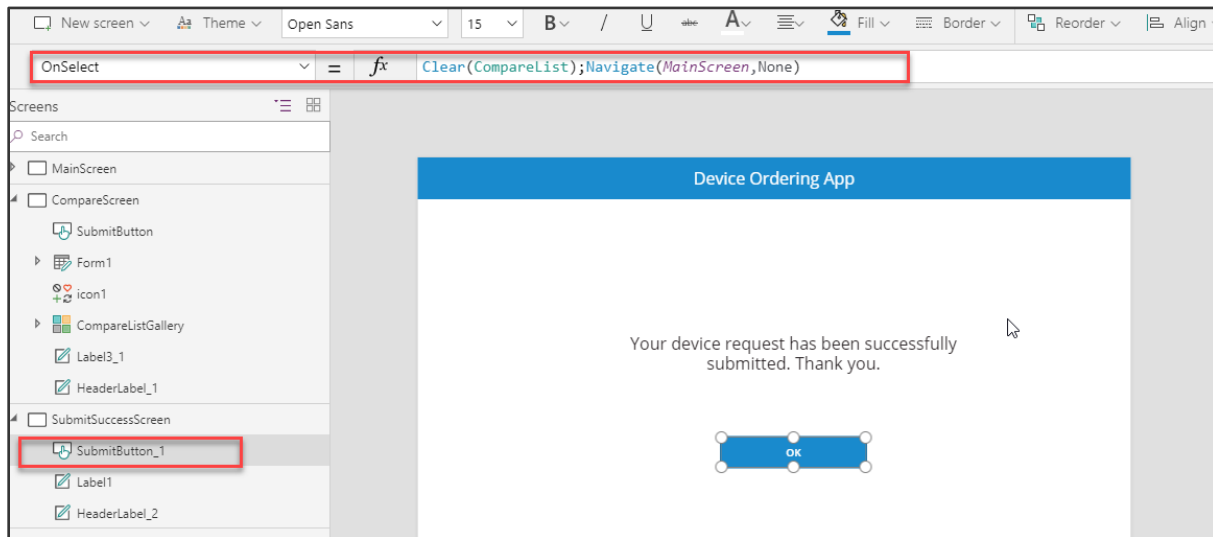
9. Add another **label** in the middle of the screen and set the **Text** to: "**Your device request has been successfully submitted. Thank you.**"
10. Increase the font size, the size of the label and center the text.



11. Add a **button** and set its **Text** to: "**OK**".
12. When the button is pressed, let us remove all the items from the CompareList collection and navigate to the first screen.

13. Set the **OnSelect** property of the button to:

```
Clear(CompareList);Navigate(MainScreen,ScreenTransition.None)
```



Note: ';' is used as a separator when multiple functions are called one after the other. If you are in a locale where ',' is used as a comma-separator, then use a double ';' here (without the single-quotes).

As a further enhancement, you can display details of the device order that was just submitted. A good way to do this is:

Add a Display Form: **Insert** -> **Form** -> **Display**

Configure its **data source** to point to the '**Device Order**' entity

Select the fields to display: Device Name, Price, Comments, Approver, Requested By, Request Date

Change the **Snap to columns** value from 3 to 1

Change the **Layout** from Vertical to Horizontal.

Set form **Item** property to `Form1.LastSubmit`

Submit another device order. Verify that the confirmation screen shows the order details.

Device Ordering App	
Your device request has been successfully submitted. Thank you.	
Device Name	Acer - Aspire S5
Price	1049.99
Comments	Current laptop does not work
Approver	admin@M365x810316.onmicrosoft.com
Requested By	admin@M365x810316.onmicrosoft.com
Request Date	4/25/2018
<div>OK</div>	

Lab survey

We would appreciate your feedback on the Power Platform technologies and on this hands-on-lab, such as the quality of documentation and the usefulness of the learning experience.

Please use the survey at <http://aka.ms/appinadayLabSurvey> to share your feedback.

You may provide feedback for each module as you complete it or at the end once you've completed all the modules. Thank you!

References

App in a Day introduces some of the key functionalities available in PowerApps, Microsoft Flow, Power BI and the Common Data Service. For an up to date list of learning references, see <http://aka.ms/powerapps-resources> and <http://aka.ms/flow-resources> and <http://powerbi.com>.

Copyright

© 2018 Microsoft Corporation. All rights reserved.

By using this demo/lab, you agree to the following terms:

The technology/functionality described in this demo/lab is provided by Microsoft Corporation for purposes of obtaining your feedback and to provide you with a learning experience. You may only use the demo/lab to evaluate such technology features and functionality and provide feedback to Microsoft. You may not use it for any other purpose. You may not modify, copy, distribute, transmit, display, perform, reproduce, publish, license, create derivative works from, transfer, or sell this demo/lab or any portion thereof.

COPYING OR REPRODUCTION OF THE DEMO/LAB (OR ANY PORTION OF IT) TO ANY OTHER SERVER OR LOCATION FOR FURTHER REPRODUCTION OR REDISTRIBUTION IS EXPRESSLY PROHIBITED.

THIS DEMO/LAB PROVIDES CERTAIN SOFTWARE TECHNOLOGY/PRODUCT FEATURES AND FUNCTIONALITY, INCLUDING POTENTIAL NEW FEATURES AND CONCEPTS, IN A SIMULATED ENVIRONMENT WITHOUT COMPLEX SET-UP OR INSTALLATION FOR THE PURPOSE DESCRIBED ABOVE. THE TECHNOLOGY/CONCEPTS REPRESENTED IN THIS DEMO/LAB MAY NOT REPRESENT FULL FEATURE FUNCTIONALITY AND MAY NOT WORK THE WAY A FINAL VERSION MAY WORK. WE ALSO MAY NOT RELEASE A FINAL VERSION OF SUCH FEATURES OR CONCEPTS. YOUR EXPERIENCE WITH USING SUCH FEATURES AND FUNCTIONALITY IN A PHYSICAL ENVIRONMENT MAY ALSO BE DIFFERENT.

FEEDBACK. If you give feedback about the technology features, functionality and/or concepts described in this demo/lab to Microsoft, you give to Microsoft, without charge, the right to use, share and commercialize your feedback in any way and for any purpose. You also give to third parties, without charge, any patent rights needed for their products, technologies and services to use or interface with any specific parts of a Microsoft software or service that includes the feedback. You will not give feedback that is subject to a license that requires Microsoft to license its software or documentation to third parties because we include your feedback in them. These rights survive this agreement.

MICROSOFT CORPORATION HEREBY DISCLAIMS ALL WARRANTIES AND CONDITIONS WITH REGARD TO THE DEMO/LAB, INCLUDING ALL WARRANTIES AND CONDITIONS OF MERCHANTABILITY, WHETHER EXPRESS, IMPLIED OR STATUTORY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. MICROSOFT DOES NOT MAKE ANY ASSURANCES OR REPRESENTATIONS WITH REGARD TO THE ACCURACY OF THE RESULTS, OUTPUT THAT DERIVES FROM USE OF DEMO/ LAB, OR SUITABILITY OF THE INFORMATION CONTAINED IN THE DEMO/LAB FOR ANY PURPOSE.

DISCLAIMER

This demo/lab contains only a portion of new features and enhancements in Microsoft PowerApps. Some of the features might change in future releases of the product. In this demo/lab, you will learn about some, but not all, new features.